



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

תוכן

הסבר סכמת הבסיס.....	2
תיאור מילולי של הטבלאות:.....	6
ERD: דיאגרמת.....	10
relational schema :תרשים.....	11
יצירה ואכלוס הטבלאות:.....	12
שאלות.....	20
Update:.....	28
Delete:.....	33
DataGenerator: הגדרת.....	37
Index.....	41
Text importer.....	43
מורחב אחרי אינטגרציה: ERD.....	48
יצירה ואכלוס טבלאות - האינטגרציה:.....	49
אילוצים.....	52
View1:.....	54
View-שאלתה 1 על ה.....	54
View2:.....	55
View-שאלתה 2 על ה.....	55
דוחות:.....	56
דו"ח 1:.....	56
דו"ח 2:.....	58
גרף 1:.....	60
גרף 2:.....	61
הרשאות גישה:.....	62
פרוצדורות:.....	64
פונקציות:.....	66
Triger:.....	68



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

[הסבר סכמת הבסיס](#)

[להלן הסכמת בסיס שעליה נתבקשנו להרחיב לצורך הפרויקט:](#)

```
create table area(areaId number(3) primary key,
    areaName varchar(20));

create table city(cityName varchar(20) primary key,
    areaId number(3) references area);

create table agent(agentID number(8) primary key,
    agentName varchar(20),
    areaID number(3) references area,
    Rating number(2),
    HireYear number(4),
    bossID number(8),
    salary float);

create table client(clientID number(8) primary key,
    clientName varchar(20),
    agentID number(8) references agent,
    phoneNr varchar(10),
    Address varchar(25),
    cityName varchar(20) references city);

create table schedule(agentId number(8),
    clientId number(8),
    meetingTime date,
    primary key (agentId, clientId, meetingTime),
    foreign key (agentId) references agent,
    foreign key (clientId) references client);
```

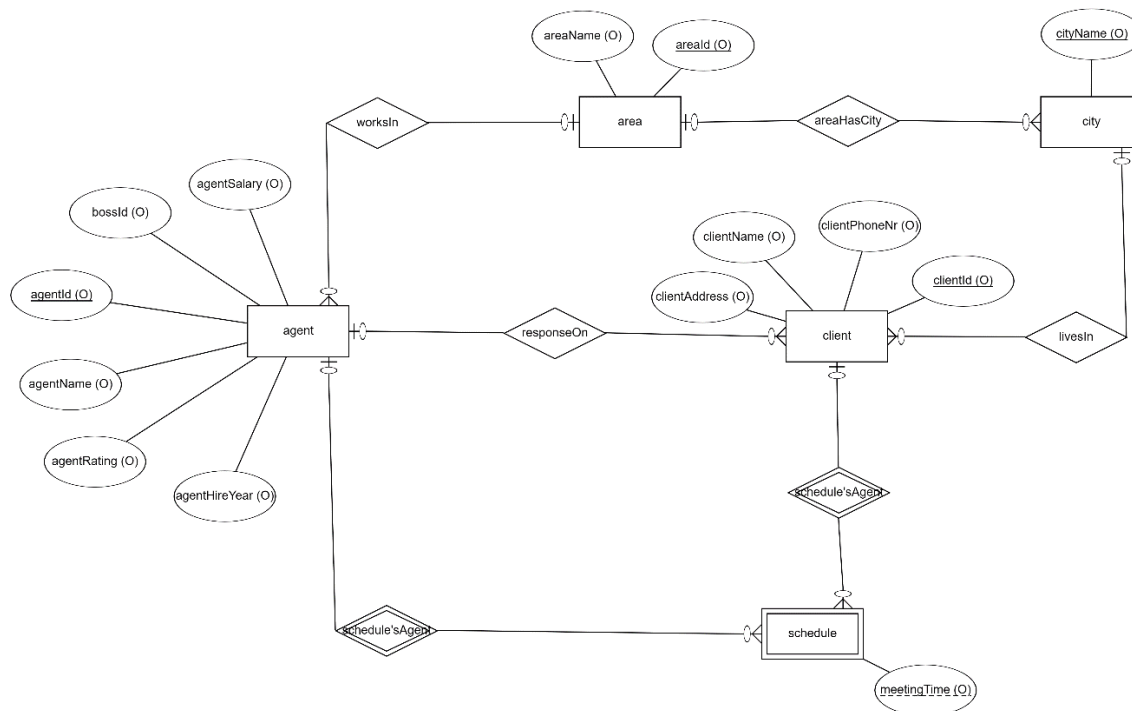


ניהול מחסן

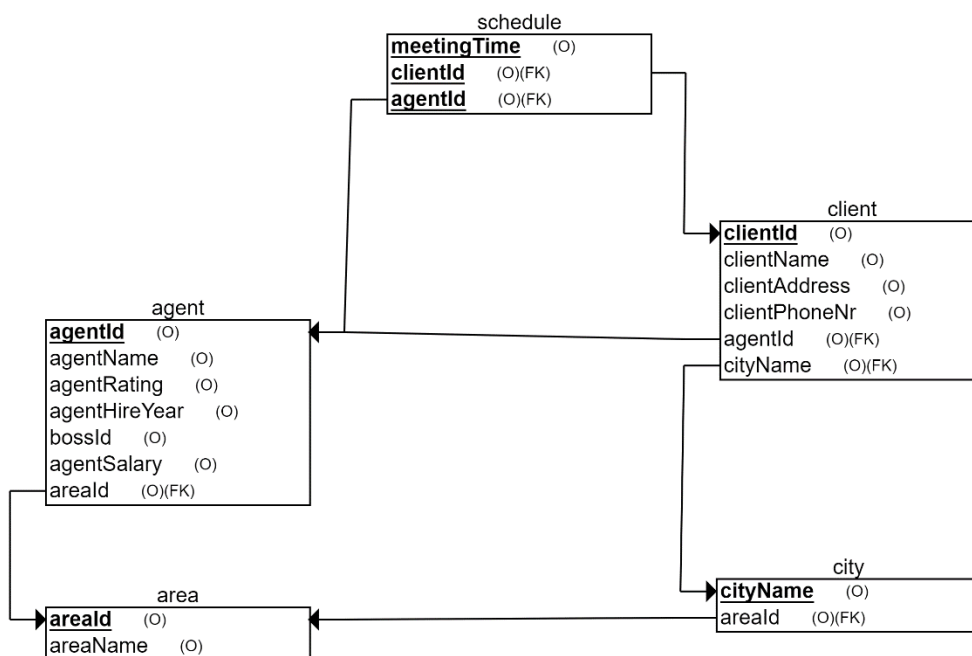
בס"ד

רחלי אסליזדה ותהילה עמרם

המרנו את הסכמה הנתונה לתרשים ERD ע"י אתר ERDPLUS ולהלן הסכימה שקיבלנו:



המרנו את התרשים שקיבלנו לסכמת קשרים:





המרנו ע"י האתר ERDPLUS לקובץ יצירת טבלאות של SQL להלן הקובץ שקיבלנו:

```
CREATE TABLE city
(
    areaId NUMERIC(3),
    cityName VARCHAR(20),
    FOREIGN KEY (areaId) REFERENCES area(areaId)
    PRIMARY KEY (cityName),
);
CREATE TABLE agent
(
    agentHireYear NUMERIC(4),
    agentName VARCHAR(20),
    agentRating NUMERIC(2),
    bossId NUMERIC(9),
    PRIMARY KEY (agentId),
    agentSalary FLOAT,
    agentId NUMERIC(9),
    areaId NUMERIC(3),
    clientId NUMERIC(9),
    FOREIGN KEY (areaId) REFERENCES area(areaId)
);
CREATE TABLE client
(
    PRIMARY KEY (clientId),
    clientName VARCHAR(20),
    clientAddress VARCHAR(25),
    agentId NUMERIC(9),
    clientPhoneNr VARCHAR(10),
    cityName VARCHAR(20),
```



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

```
(  
    FOREIGN KEY (agentId) REFERENCES agent(agentId),  
    FOREIGN KEY (cityName) REFERENCES city(cityName)  
    FOREIGN KEY (agentId) REFERENCES agent(agentId)  
);  
CREATE TABLE schedule  
    meetingTime DATE,  
    clientId NUMERIC(9),  
    agentId NUMERIC(9),  
    PRIMARY KEY (meetingTime, clientId, agentId),  
    FOREIGN KEY (clientId) REFERENCES client(clientId),  
);
```

ניתן לראות שקיבלנו את אותו קובץ CreateTable כמו הנתון.

בחרנו בנושא של ניהול מחסני החנויות של "אושר עד" והתבקשנו להרחיב את סכמת הבסיס בשישה ישויות נוספות.

בפרויקט ניבנה בסיס נתונים לניהול הזמנות. לחברה ישנם מספר סניפים בארץ, המוצרים מחולקים לפי מחלקות- קטגוריות.

לכל הזמנה של סניף יכולים להיות מספר מוצרים, לקוח יכול להחזיר מוצר וכו..



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם
תיאור מילולי של הטבלאות:

city (cityName, areald).

company (companyId, companyName, companyCountry, companyEstablishedYear).

category (categoryId, categoryName).

branch (branchId, branchName, cityName).

agent (agentId, agentName, agentRating, agentHireYear, bossId, agentSalary).

orderFromBasement (orderNumber, orderDate, branchId, agentId).

product (productBarcode, productName, categoryId, companyId, productPrice).

represent (companyId, agentId).

orderedProduct (productBarcode, orderNumber , orderAmount , expiryDate ,soldAmount).

client (clientId, clientName, clientAddress, clientPhoneNr).

area (areald, areaName).

schedule (meetingTime , clientId, agentId).

returns (returnsId, returnsReason, returnsDate, productBarcode).



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

<u>Table Name:</u>	<u>Attribute:</u>	<u>Explanation</u>
city	<u>cityName</u>	מפתח
	<u>areald</u>	מפתח זר
client	<u>clientId</u>	מפתח
	clientName	
	clientAddress	
	clientPhoneNr	
agent	<u>agentId</u>	מפתח
	agentName	
	agentRating	
	agentHireYear	
	bossId	
	agentSalary	
area	<u>areald</u>	מפתח
	areaName	
schedule	<u>meetingTime</u>	מפתח
	<u>clientId</u>	מפתח
	<u>agentId</u>	מפתח



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

company	<u>companyId</u>	מפתח
	companyName	שם חברה
	companyCountry	ארץ מוצא
	companyEstablishedYear	שנת היווסדות
category	<u>categoryId</u>	מפתח
	categoryName	שם קטגוריה
product	<u>productBarcode</u>	מפתח
	productName	שם מוצר
	productPrice	מחיר מוצר
	<u>categoryId</u>	מפתח זר מטבלת category
	<u>companyId</u>	מפתח זר מטבלת company
represent	<u>companyId</u>	מפתח זר מטבלת company
	<u>agentId</u>	מפתח זר מטבלת agent
branch	<u>branchId</u>	מפתח
	branchName	שם סניף
	<u>cityName</u>	מפתח זר מטבלת city
orderFromBasement	<u>orderNumber</u>	מפתח



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

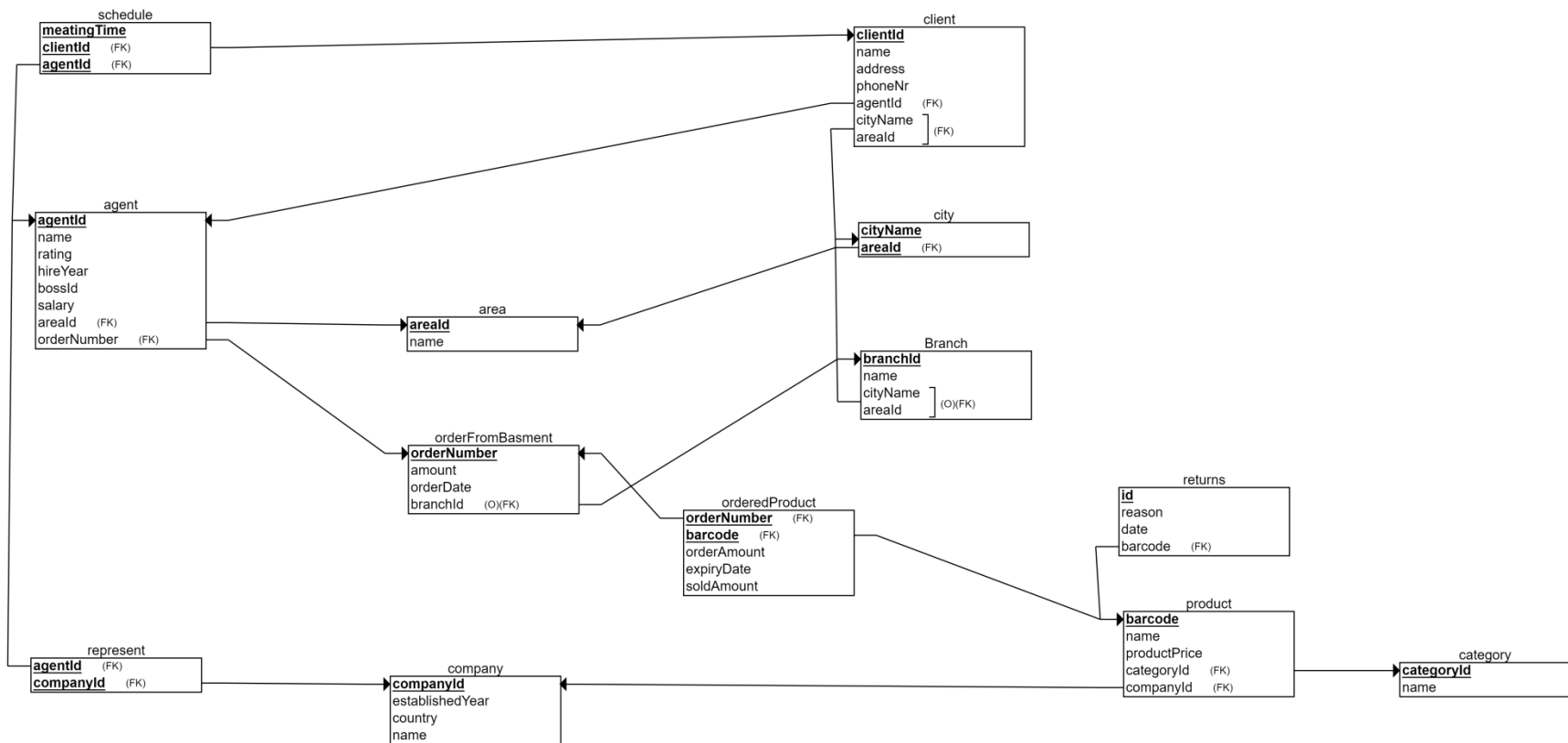
	orderDate	תאריך הזמנה
	<u>branchId</u>	מפתח זר מטבלת branch
	<u>agentId</u>	מפתח זר מטבלת agent
orderedProduct	<u>productBarcode</u>	מפתח זר מטבלת product
	<u>orderNumber</u>	מפתח זר מטבלת orderFromBasement
	orderAmount	כמות הזמנה
	expiryDate	תאריך תפוגה
	soldAmount	כמות שנמכרה
returns	<u>returnsId</u>	מפתח
	returnsReason	סיבת ההחזרה
	returnsDate	תאריך החזרת מוצר
	<u>productBarcode</u>	מפתח זר מטבלת product

ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

[תרשים relational schema](#):





ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

יצירה ואכלוס הטבלאות:

Company

יצירת טבלה:

```
CREATE TABLE company
(
    companyId NUMERIC(9),
    companyName VARCHAR(30),
    companyCountry VARCHAR(30),
    companyEstablishedYear DATE,
    PRIMARY KEY (companyId)
);
```

הכנסת נתונים:

-- company insert

```
INSERT INTO company (companyId, companyName, companyCountry, companyEstablishedYear)
VALUES (001, 'Pilot', 'China', to_date('22/06/2017 14:00', 'DD/MM/YYYY HH24:MI'));
```

```
INSERT INTO company (companyId, companyName, companyCountry, companyEstablishedYear)
VALUES (002, 'Tiv Taam', 'Israel', to_date('20/06/2018 14:00', 'DD/MM/YYYY HH24:MI'));
```

```
INSERT INTO company (companyId, companyName, companyCountry, companyEstablishedYear)
VALUES (003, 'Osem', 'Israel', to_date('22/06/2017 14:00', 'DD/MM/YYYY HH24:MI'));
```

```
INSERT INTO company (companyId, companyName, companyCountry, companyEstablishedYear)
VALUES (004, 'Hanasich', 'Israel', to_date('22/06/2017 14:00', 'DD/MM/YYYY HH24:MI'));
```

```
INSERT INTO company (companyId, companyName, companyCountry, companyEstablishedYear)
VALUES (005, 'Taaman', 'Israel', to_date('22/06/2017 14:00', 'DD/MM/YYYY HH24:MI'));
```

הצגת הנתונים:

	COMPANYID	COMPANYNAME	COMPANYCOUNTRY	COMPANYESTABLISHEDYEAR
1	1	Pilot	China	22/06/2017 14:00:00
2	2	Tiv Taam	Israel	20/06/2018 14:00:00
3	3	Osem	Israel	22/06/2017 14:00:00
4	4	Hanasich	Israel	22/06/2017 14:00:00
5	5	Taaman	Israel	22/06/2017 14:00:00



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

represent

יצירת טבלה:

```
CREATE TABLE represent
(
  companyId NUMERIC(9) NOT NULL,
  agentId NUMERIC(9) NOT NULL,
  PRIMARY KEY (companyId, agentId),
  FOREIGN KEY (companyId) REFERENCES company(companyId),
  FOREIGN KEY (agentId) REFERENCES agent(agentId)
);
```

הכנסת נתונים:

```
-- represent insert
INSERT INTO represent (companyId, agentId)
VALUES (001, 660084321);

INSERT INTO represent (companyId, agentId)
VALUES (002, 724732338);

INSERT INTO represent (companyId, agentId)
VALUES (003, 683733405);

INSERT INTO represent (companyId, agentId)
VALUES (004, 615988674);

INSERT INTO represent (companyId, agentId)
VALUES (005, 564441324);
```

הצגת הנתונים:

	COMPANYID	AGENTID
1	1	660084321
2	2	724732338
3	3	683733405
4	4	615988674
5	5	564441324



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

category

יצירת טבלה:

```
CREATE TABLE category
(
    categoryId NUMERIC(9),
    categoryName VARCHAR(20),
    PRIMARY KEY (categoryId)
);
```

הכנסת נתונים:

```
-- category insert

INSERT INTO category (categoryId, categoryName)
VALUES (001, 'frozen');

INSERT INTO category (categoryId, categoryName)
VALUES (002, 'vagetables');

INSERT INTO category (categoryId, categoryName)
VALUES (003, 'baking');

INSERT INTO category (categoryId, categoryName)
VALUES (004, 'cleanliness');

INSERT INTO category (categoryId, categoryName)
VALUES (005, 'canned food');
```

הצגת הנתונים:

	CATEGORYID	CATEGORYNAME	
1	1	frozen	...
2	2	vagetables	...
3	3	baking	...
4	4	cleanliness	...
5	5	canned food	...



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

product

יצירת טבלה:

```
CREATE TABLE product
(
    productBarcode NUMERIC(10),
    productName CHAR(30),
    categoryId NUMERIC(9),
    companyId NUMERIC(9),
    productPrice FLOAT,
    PRIMARY KEY (productBarcode),
    FOREIGN KEY (categoryId) REFERENCES category(categoryId)
    FOREIGN KEY (companyId) REFERENCES company(companyId)
);
```

הכנסת נתונים:

```
-- product insert
INSERT INTO product (productBarcode, productName, categoryId, productPrice, companyId)
VALUES (001, 'corn', 005, 40, 5);

INSERT INTO product (productBarcode, productName, categoryId, productPrice, companyId)
VALUES (002, 'soy sausages', 001, 20, 4);

INSERT INTO product (productBarcode, productName, categoryId, productPrice, companyId)
VALUES (003, 'bleach', 004, 34, 3);

INSERT INTO product (productBarcode, productName, categoryId, productPrice, companyId)
VALUES (004, 'flour', 003, 2.23, 2);

INSERT INTO product (productBarcode, productName, categoryId, productPrice, companyId)
VALUES (005, 'tomato', 002, 8, 1);
```

הצגת נתונים:

	PRODUCTBARCODE	PRODUCTNAME	CATEGORYID	COMPANYID	PRODUCTPRICE	
1	1	corn	...	5	5	40
2	2	soy sausages	...	1	4	20
3	3	bleach	...	4	3	34
4	4	flour	...	3	2	2.23
5	5	tomato	...	2	1	8



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

Branch

יצירת טבלה:

```
CREATE TABLE branch
(
    branchId NUMERIC(9),
    branchName VARCHAR(30),
    cityName VARCHAR(20),
    PRIMARY KEY (branchId),
    FOREIGN KEY (cityName) REFERENCES city(cityName)
);
```

הכנסת נתונים:

```
-- branch insert
INSERT INTO branch (branchId, branchName, cityName)
VALUES (1, 'Rishon LeTzion hagadol', 'Rishon LeTzion');

INSERT INTO branch (branchId, branchName, cityName)
VALUES (2, 'Nahariya hagadol', 'Nahariya');

INSERT INTO branch (branchId, branchName, cityName)
VALUES (3, 'Nazareth hagadol', 'Nazareth');

INSERT INTO branch (branchId, branchName, cityName)
VALUES (4, 'Nazareth Illit hagadol', 'Nazareth Illit');

INSERT INTO branch (branchId, branchName, cityName)
VALUES (5, 'Nesher hagadol', 'Nesher');
```

הצגת הנתונים:

	BRANCHID	BRANCHNAME	CITYNAME
▶ 1	1	Rishon LeTzion hagadol	Rishon LeTzion
2	2	Nahariya hagadol	Nahariya
3	3	Nazareth hagadol	Nazareth
4	4	Nazareth Illit hagadol	Nazareth Illit
5	5	Nesher hagadol	Nesher



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

orderFromBasement

יצירת טבלה:

```
CREATE TABLE orderFromBasement
(
    orderNumber NUMERIC(9),
    branchId NUMERIC(9),
    agentId NUMERIC(9),
    orderDate DATE,
    FOREIGN KEY (agentId) REFERENCES agent(agentId),
    PRIMARY KEY (orderNumber),
    FOREIGN KEY (branchId) REFERENCES branch(branchId)
);
```

הכנסת הנתונים:

```
-- orderFromBasement insert
INSERT INTO orderFromBasement (orderNumber,branchId,agentId,orderDate)
VALUES (1,1,660084321,to_date('20/06/2018 14:00', 'DD/MM/YYYY HH24:MI'));

INSERT INTO orderFromBasement (orderNumber,branchId,agentId,orderDate)
VALUES (2,2,724732338,to_date('20/06/2017 14:00', 'DD/MM/YYYY HH24:MI'));

INSERT INTO orderFromBasement (orderNumber,branchId,agentId,orderDate)
VALUES (3,3,683733405,to_date('20/06/2018 14:00', 'DD/MM/YYYY HH24:MI'));

INSERT INTO orderFromBasement (orderNumber,branchId,agentId,orderDate)
VALUES (4,4,615988674,to_date('20/06/2018 14:00', 'DD/MM/YYYY HH24:MI'));

INSERT INTO orderFromBasement (orderNumber,branchId,agentId,orderDate)
VALUES (5,5,564441324,to_date('20/06/2018 14:00', 'DD/MM/YYYY HH24:MI'));
```

הצגת הנתונים:

	ORDERNUMBER	BRANCHID	AGENTID	ORDERDATE	
1	1	1	660084321	20/06/2018 14:00:00	▼
2	2	2	724732338	20/06/2017 14:00:00	▼
3	3	3	683733405	20/06/2018 14:00:00	▼
4	4	4	615988674	20/06/2018 14:00:00	▼
5	5	5	564441324	20/06/2018 14:00:00	▼



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

orderedProduct

יצירת טבלה:

```
CREATE TABLE orderedProduct
(
    productBarcode NUMERIC(10) NOT NULL,
    orderNumber NUMERIC(9) NOT NULL,
    orderAmount INT,
    EXPIRYDATE date,
    SOLDAMOUNT INT,
    PRIMARY KEY (productBarcode, orderNumber),
    FOREIGN KEY (productBarcode) REFERENCES product (productBarcode),
    FOREIGN KEY (orderNumber) REFERENCES orderFromBasement (orderNumber)
);
```

הכנסת נתונים:

```
-- orderedProduct insert
INSERT INTO orderedProduct (productBarcode, orderNumber,orderAmount,expiryDate, soldAmount)
VALUES (001,5,100,to_date('20/06/2018 14:00', 'DD/MM/YYYY HH24:MI'),30);

INSERT INTO orderedProduct (productBarcode, orderNumber,orderAmount,expiryDate, soldAmount)
VALUES (002,4,500,to_date('13/09/2010 14:00', 'DD/MM/YYYY HH24:MI'),500);

INSERT INTO orderedProduct (productBarcode, orderNumber,orderAmount,expiryDate, soldAmount)
VALUES (003,1,380,to_date('01/01/2017 14:00', 'DD/MM/YYYY HH24:MI'),90);

INSERT INTO orderedProduct (productBarcode, orderNumber,orderAmount,expiryDate, soldAmount)
VALUES (004,3,786,to_date('12/05/2015 14:00', 'DD/MM/YYYY HH24:MI'),500);

INSERT INTO orderedProduct (productBarcode, orderNumber,orderAmount,expiryDate, soldAmount)
VALUES (005,2,98,to_date('20/10/2000 14:00', 'DD/MM/YYYY HH24:MI'),97);
```

הצגת הנתונים:

	PRODUCTBARCODE	ORDERNUMBER	ORDERAMOUNT	EXPIRYDATE	SOLDAMOUNT
1	1	5	100	20/06/2018 14:00:00 ▾	30
2	2	4	500	13/09/2010 14:00:00 ▾	500
3	3	1	380	01/01/2017 14:00:00 ▾	90
4	4	3	786	12/05/2015 14:00:00 ▾	500
5	5	2	98	20/10/2000 14:00:00 ▾	97



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

returns

יצירת טבלה:

```
CREATE TABLE returns
(
    returnsReason VARCHAR(500),
    returnsId NUMERIC(9),
    returnsDate DATE,
    productBarcode NUMERIC(10),
    PRIMARY KEY (returnsId),
    FOREIGN KEY (productBarcode) REFERENCES product (productBarcode)
);
```

הכנסת נתונים:

```
-- returns insert
INSERT INTO returns (productBarcode, returnsReason, returnsId, returnsDate)
VALUES (001, 'defective', 1, to_date('20/06/2018 14:00', 'DD/MM/YYYY HH24:MI'));

INSERT INTO returns (productBarcode, returnsReason, returnsId, returnsDate)
VALUES (002, 'rotten', 2, to_date('20/06/2018 14:00', 'DD/MM/YYYY HH24:MI'));

INSERT INTO returns (productBarcode, returnsReason, returnsId, returnsDate)
VALUES (003, 'bad smell', 3, to_date('20/06/2018 14:00', 'DD/MM/YYYY HH24:MI'));

INSERT INTO returns (productBarcode, returnsReason, returnsId, returnsDate)
VALUES (004, 'opened', 4, to_date('20/06/2018 14:00', 'DD/MM/YYYY HH24:MI'));

INSERT INTO returns (productBarcode, returnsReason, returnsId, returnsDate)
VALUES (005, 'broken', 5, to_date('20/06/2018 14:00', 'DD/MM/YYYY HH24:MI'));
```

הצגת הנתונים:

	RETURNSREASON	RETURNSID	RETURNSDATE	PRODUCTBARCODE
1	defective	1	20/06/2018 14:00:00	1
2	rotten	2	20/06/2018 14:00:00	2
3	bad smell	3	20/06/2018 14:00:00	3
4	opened	4	20/06/2018 14:00:00	4
5	broken	5	20/06/2018 14:00:00	5



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

שאלות

שאלת 1:

שאלתה עם פרמטרים, רוצים לדעת אילו פריטים הוחזרו בתקופת זמן.

המשתמש מזין את הטווח הרצוי והוא מקבל כפלט את הפריטים שהוחזרו באותה תקופת זמן.

```
-- query 1 with params
SELECT productBarcode AS "Barcode", productName AS "Name", companyName AS "Made By", COUNT(returnsId) AS "Amount", returnsDate AS "Return Date"
FROM product p NATURAL JOIN company c NATURAL JOIN returns r
where
r.returnsDate BETWEEN to_date(&d_from, 'DD/MM/YYYY HH24:MI') AND to_date(&d_to, 'DD/MM/YYYY HH24:MI')
GROUP BY returnsId, productBarcode, productName, companyName, returnsDate;
```

Name	Value
d_from	'1/1/2000 14:00'
d_to	'2/2/2020 12:00'

	Barcode	Name	Made By	Amount	Return Date
1	1	corn	Taaman	1	20/06/2018 14:00:00
2	2	soy sausages	Hanasich	1	20/06/2018 14:00:00
3	5	tomato	Pilot	1	20/06/2018 14:00:00
4	3	bleach	Osem	1	20/06/2018 14:00:00
5	4	flour	Tiv Taam	1	20/06/2018 14:00:00



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

שאלתה 2:

סוכם את כמות המוצרים להזמנה

```
-- query 2
SELECT * from
(SELECT sum(op.orderamount) as "amout of products", ordernumber
FROM ORDEREDPRODUCT op NATURAL JOIN ORDERFROMBASEMENT OFB
group by ordernumber) maxresults;
```

	amout of products	ORDERNUMBER
1	380	1
2	98	2
3	100	5
4	500	4
5	786	3



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

שאלתה 3:

בודקת מהו המוצר הנמכר ביותר

```
-- query 3
SELECT query1.*
FROM (SELECT productBarcode as "barcode", SUM(orderedproduct.orderamount) AS total_amt, productName as "name"
      FROM orderedproduct natural join product
      GROUP BY productbarcode, productName) query1,
      (SELECT MAX(query2.total_amt) AS highest_amt
      FROM (SELECT productBarcode, SUM(orderedproduct.orderamount) AS total_amt
            FROM orderedproduct
            GROUP BY orderedproduct.productbarcode) query2) query3
WHERE query1.total_amt = query3.highest_amt;
```

	barcode	TOTAL_AMT	name	
1	4	786	flour	...



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

שאלתה 4:

שאלתה עם פרמטרים, המשתמש מזין כמות ימים עד לתאריך תפוגה והפלט: המוצרים שתוקפם יעבור תוך כמות הימים שהמשמש הזין.

-- query 4 with params

```
SELECT branchName as "branch name", cityName as "city", productName as "product name", productBarcode as "barcode", expiryDate as "expiry date"
from orderedproduct natural join orderfrombasement natural join branch natural join product
where (SYSDATE - expiryDate) < <name="days till expiry" type="integer" hint="enter integer value"> and SYSDATE >= expiryDate;
```

	branch name	city	product name	barcode	expiry date
1	Nesher hagadol	Nesher	corn	1	20/06/2018 14:00:00



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

שאלתה 5:

המשתמש בוחר סניף ומוצרים, וזה מציג לו כמה יש במלאי.

```
-- query 5 with params
select * from (SELECT sum(orderedproduct.orderamount-orderedproduct.soldamount) AS "in stock",productbarcode
FROM orderedproduct natural join orderfrombasement natural join branch
where branchid = {<name="branch" list="select branchid, branchname from branch order by branchname" description="yes" restricted="yes">
GROUP BY ordernumber,productbarcode) natural join product
where productbarcode in ({<name="product" list="select productbarcode, productname from product order by productname" description="yes" multiselect="yes">);
```

Name	Value
branch	Nesher hagadol
product	corn,tomato

OK Cancel Clear

	PRODUCTBARCODE	in stock	PRODUCTNAME	CATEGORYID	COMPANYID	PRODUCTPRICE
1	1	70	corn	5	5	40



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

שאלתה 6:

מציג איזה קטגוריה הכי פופולארית מבין המוצרים.

```
-- query 6
SELECT query1.*
FROM (SELECT categoryid, SUM(orderamount) AS total_amt, categoryName
      FROM orderedproduct natural join product natural join category
      GROUP BY categoryid, categoryName ) query1,
      (SELECT MAX(query2.total_amt) AS highest_amt
      FROM (SELECT categoryid, SUM(orderamount) AS total_amt
            FROM orderedproduct natural join product natural join category
            GROUP BY categoryid) query2) query3
WHERE query1.total_amt = query3.highest_amt;
```

	CATEGORYID	TOTAL_AMT	CATEGORYNAME	
1	3	786	baking	...



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

שאלתה 7:

שאלתה עם פרמטרים, המשתמש בוחר סניף ומוצג כל היסטורית ההזמנות של אותו הסניף.

```
select * from  
(SELECT SUM(orderedproduct.orderamount) AS total_amt, ordernumber, orderdate, agentId  
FROM orderedproduct natural join orderfrombasement  
where branchid = {<name="branch" list="select branchid, branchname from branch order by branchname" description="yes" restricted="yes">  
GROUP BY ordernumber,orderdate, agentId) query1
```

Name	Value
branch	Rishon LeTzion hagadol

	TOTAL_AMT	ORDERNUMBER	ORDERDATE	AGENTID
1	380	1	20/06/2018 14:00:00	660084321



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

שאלתה 8:

שאלתה שמחזירה את המוצרים הלא מוחזרים מהחברה.
זאת אומרת, עד כמה אני מרוצה מהחברה (תלוי במספר החזרות המוצרים).

```
-- query 8
SELECT 100*(query1.s-query2.c)*1.0/query1.s as "satisfaction rates", query1.companyname as "company"
FROM (select sum(orderamount) as s, companyname,companyid
from orderedproduct natural join product natural join company
group by companyid,companyname) query1,
(select count(productbarcode) as c, companyid, companyname
from returns natural join product natural join company
group by companyid,companyname) query2
WHERE query1.companyid = query2.companyid and query1.s>0
order by 1 desc;
```

	satisfaction rates	company	
1	99.8727735368957	Tiv Taam	...
2	99.8	Hanasich	...
3	99.7368421052632	Osem	...
4	99	Taaman	...
5	98.9795918367347	Pilot	...



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

[:Update](#)

[עדכון מספר 1:](#)

השאלתה מעדכנת את כל הרשומות בטבלה Returns שהערך של סיבת ההחזרה ריק , להיות not known.

	RETURNSREASON	RETURNSID	RETURNSDATE	PRODUCTBARCODE
1	defective	...	1 20/06/2018 14:00:00 ▼	1
2	rotten	...	2 20/06/2018 14:00:00 ▼	2
3	bad smell	...	3 20/06/2018 14:00:00 ▼	3
4	opened	...	4 20/06/2018 14:00:00 ▼	4
5		...	5 20/06/2018 14:00:00 ▼	5

```
update returns
set returnsreason= 'not known'
where returnsreason is null;
```

	RETURNSREASON	RETURNSID	RETURNSDATE	PRODUCTBARCODE
1	defective	...	1 20/06/2018 14:00:00 ▼	1
2	rotten	...	2 20/06/2018 14:00:00 ▼	2
3	bad smell	...	3 20/06/2018 14:00:00 ▼	3
4	opened	...	4 20/06/2018 14:00:00 ▼	4
5	not known	...	5 20/06/2018 14:00:00 ▼	5

```
rollback;
```

לאחר פעולה זו מצב הטבלה יחזור להיות כמו לפני העדכון.

אם ביצענו commit אזי ה-rollback יחזיר עד ל-commit האחרון כלומר מאחרי ה-commit החזרה אחורה לא תעבוד



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

עדכון מספר 2:

כל המוצרים שנמכרו יותר מהכמות שהוזמנו מעודכנים שהכמות שנמכרה שווה לכמות שהוזמנה.

	PRODUCTBARCODE	ORDERNUMBER	ORDERAMOUNT	EXPIRYDATE	SOLDAMOUNT
1	1	5	100	20/06/2018 14:00:00 ▾	30
2	2	4	500	13/09/2010 14:00:00 ▾	500
3	3	1	380	01/01/2017 14:00:00 ▾	7000
4	4	3	786	12/05/2015 14:00:00 ▾	500
5	5	2	98	20/10/2000 14:00:00 ▾	97

```
update orderedproduct  
set soldamount = orderamount  
where soldamount > orderamount;  
... .
```

	PRODUCTBARCODE	ORDERNUMBER	ORDERAMOUNT	EXPIRYDATE	SOLDAMOUNT
1	1	5	100	20/06/2018 14:00:00 ▾	30
2	2	4	500	13/09/2010 14:00:00 ▾	500
3	3	1	380	01/01/2017 14:00:00 ▾	380
4	4	3	786	12/05/2015 14:00:00 ▾	500
5	5	2	98	20/10/2000 14:00:00 ▾	97



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

עדכון מספר 3:

לוקחת את כל ההזמנות בתאריך הנוכחי ובודקת אם הוזמן בהם מוצר שהברקוד שלו שווה ל-2.

אם כן, היא דוחה את תאריך ההזמנה בשבוע.

	ORDERNUMBER	BRANCHID	AGENTID	ORDERDATE
1	7	5	564441324	14/05/2019 14:00:00
2	1	1	660084321	20/06/2018 14:00:00
3	2	2	724732338	20/06/2017 14:00:00
4	3	3	683733405	20/06/2018 14:00:00
5	4	4	615988674	20/06/2018 14:00:00
6	5	5	564441324	20/06/2018 14:00:00
7	6	5	564441324	20/06/2018 14:00:00

	PRODUCTBARCODE	ORDERNUMBER	ORDERAMOUNT	EXPIRYDATE	SOLDAMOUNT
1	2	7	98	14/05/2019 14:00:00	97
2	1	5	100	20/06/2018 14:00:00	30
3	2	4	500	13/09/2010 14:00:00	500
4	3	1	380	01/01/2017 14:00:00	90
5	4	3	786	12/05/2015 14:00:00	500
6	5	2	98	20/10/2000 14:00:00	97

```

UPDATE orderfrombasement
SET orderdate = orderdate+7
WHERE EXISTS (SELECT productbarcode
FROM orderedproduct
WHERE productbarcode = 2)
and (orderdate -sysdate<1 or
sysdate- orderdate<1) ;

```

	ORDERNUMBER	BRANCHID	AGENTID	ORDERDATE
1	7	5	564441324	21/05/2019 14:00:00
2	1	1	660084321	27/06/2018 14:00:00
3	2	2	724732338	27/06/2017 14:00:00
4	3	3	683733405	27/06/2018 14:00:00
5	4	4	615988674	27/06/2018 14:00:00
6	5	5	564441324	27/06/2018 14:00:00
7	6	5	564441324	27/06/2018 14:00:00



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

עדכון מספר 4:

השאלתה בודקת את כל המוצרים שלא נמכרו עד כה ומורידה את מחירם בעשרה אחוזים.

	PRODUCTBARCODE	PRODUCTNAME	CATEGORYID	COMPANYID	PRODUCTPRICE	
1	1	corn	...	5	5	40
2	2	soy sausages	...	1	4	20
3	3	bleach	...	4	3	34
4	4	flour	...	3	2	2.23
5	5	tomato	...	2	1	8
6	6	cucumber	...	2	1	8

	PRODUCTBARCODE	ORDERNUMBER	ORDERAMOUNT	EXPIRYDATE	SOLDAMOUNT
1	2	7	98	14/05/2019 14:00:00 ▼	97
2	1	5	100	20/06/2018 14:00:00 ▼	30
3	2	4	500	13/09/2010 14:00:00 ▼	500
4	3	1	380	01/01/2017 14:00:00 ▼	90
5	4	3	786	12/05/2015 14:00:00 ▼	500
6	5	2	98	20/10/2000 14:00:00 ▼	97

```
UPDATE product
set productprice = productprice*0.9
where not exists (select productbarcode from orderedproduct where product.productbarcode=orderedproduct.productbarcode);
```

	PRODUCTBARCODE	PRODUCTNAME	CATEGORYID	COMPANYID	PRODUCTPRICE
1	1	corn	5	5	40
2	2	soy sausages	1	4	20
3	3	bleach	4	3	34
4	4	flour	3	2	2.23
5	5	tomato	2	1	8
6	6	cucamber	2	1	7.2



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

[Delete:](#)

[מחיקה מספר 1:](#)

מוחק את כל הקטגוריות שהשם שלהם הוא פחות משלוש אותיות.

	CATEGORYID	CATEGORYNAME	
1	1	frozen	...
2	2	vegetables	...
3	3	baking	...
4	4	cleanliness	...
5	5	canned food	...
6	6	df	...

```
delete from category
where length(categoryName) < 3;
```

	CATEGORYID	CATEGORYNAME	
1	1	frozen	...
2	2	vegetables	...
3	3	baking	...
4	4	cleanliness	...
5	5	canned food	...



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

מחיקה מספר 2:

מוחק את כל הפריטים שתאריך ההחזרה שלהם גדול מהתאריך הנוכחי.

	RETURNSREASON		RETURNSID	RETURNSDATE		PRODUCTBARCODE	
1	defective	...	1	20/06/2018 14:00:00	▼		1
2	rotten	...	2	20/06/2018 14:00:00	▼		2
3	bad smell	...	3	20/06/2018 14:00:00	▼		3
4	opened	...	4	20/06/2018 14:00:00	▼		4
5	not known	...	5	20/06/2018 14:00:00	▼		5
6		...	6	25/04/2019	▼		3

```
delete from returns
where returnsdate > sysdate;
```

	RETURNSREASON		RETURNSID	RETURNSDATE		PRODUCTBARCODE	
1	defective	...	1	20/06/2018 14:00:00	▼		1
2	rotten	...	2	20/06/2018 14:00:00	▼		2
3	bad smell	...	3	20/06/2018 14:00:00	▼		3
4	opened	...	4	20/06/2018 14:00:00	▼		4
5	not known	...	5	20/06/2018 14:00:00	▼		5



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

מחיקה מספר 3:

מחיקת כל הקטגוריות שאין להן מוצרים.

להלן רשימת הקטגוריות הקיימות בבסיס הנתונים:

	CATEGORYID	CATEGORYNAME	
1	1	frozen	...
2	2	vegetables	...
3	3	baking	...
4	4	cleanliness	...
5	5	canned food	...
6	6	fn	...

מספר קטגוריה וכמה מוצרים יש ממנה:

	C	CATEGORYID
1	1	1
2	1	2
3	1	5
4	1	4
5	1	3

```
delete from category c
where
c.categoryid not in
(select query1.categoryid from ( select count(categoryid) as c,categoryid from product group by categoryid) query1)
```

רשימת הקטגוריות בבסיס הנתונים אחרי הרצת השאילתה:

	CATEGORYID	CATEGORYNAME	
1	1	frozen	...
2	2	vegetables	...
3	3	baking	...
4	4	cleanliness	...
5	5	canned food	...



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

מחיקה מספר 4:

מוחק את כל ההזמנות בלי מוצרים בתאריך מסויים.

להלן רשימת הקטגוריות הקיימות בבסיס הנתונים:

	ORDERNUMBER	BRANCHID	AGENTID	ORDERDATE
1	1	1	660084321	20/06/2018 14:00:00
2	2	2	724732338	20/06/2017 14:00:00
3	3	3	683733405	20/06/2018 14:00:00
4	4	4	615988674	20/06/2018 14:00:00
5	5	5	564441324	20/06/2018 14:00:00
6	6	5	564441324	20/06/2018 14:00:00

מספר קטגוריה וכמה מוצרים יש ממנה:

	C	ORDERNUMBER
1	1	1
2	1	2
3	1	5
4	1	4
5	1	3

```
delete from orderfrombasement ofb
where
ofb.ordernumber not in
(select query1.ordernumber from ( select count(ordernumber) as c,ordernumber from orderedproduct group by ordernumber) query1)
```

רשימת הקטגוריות בבסיס הנתונים אחרי הרצת השאילתה:

	ORDERNUMBER	BRANCHID	AGENTID	ORDERDATE
1	1	1	660084321	20/06/2018 14:00:00
2	2	2	724732338	20/06/2017 14:00:00
3	3	3	683733405	20/06/2018 14:00:00
4	4	4	615988674	20/06/2018 14:00:00
5	5	5	564441324	20/06/2018 14:00:00



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

: [DataGenerator](#) הגדרת

Category

זמן ריצה: 10.36 שניות.

CATEGORY

Owner	Table	Number of records
TAMRAM	CATEGORY	3000..3000

Name	Type	Size	Data
CATEGORYID	NUMBER	9	Sequence(10, [Inc], [WithinParent])
CATEGORYNAME	VARCHAR2	20	Elements.Name
*			

Company

זמן ריצה: 12.812 שניות.

COMPANY

Owner	Table	Number of records
TAMRAM	COMPANY	3000..3000

Name	Type	Size	Data
COMPANYID	NUMBER	9	Sequence(10, [Inc], [WithinParent])
COMPANYNAME	VARCHAR2	30	Company
COMPANYCOUNTRY	VARCHAR2	30	Country
COMPANYESTABLISHED	DATE		Random(1/1/1900, 1/1/2000)
...			



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

Branch

זמן ריצה: 10.344 שניות.

BRANCH

Owner

Table

Number of records

TAMRAM

BRANCH

3000..3000

Name	Type	Size	Data
BRANCHID	NUMBER	9	Sequence(10, [Inc], [WithinParent])
BRANCHNAME	VARCHAR2	30	Elements.Name
CITYNAME	VARCHAR2	20	List(select cityName from city)
*			

Represent

זמן ריצה: 174.656

REPRESENT

Owner

Table

Number of records

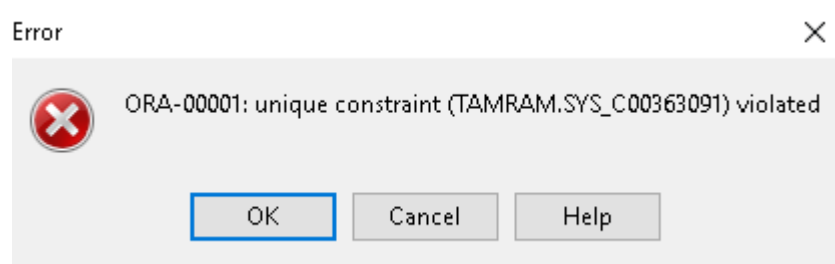
TAMRAM

REPRESENT

3000..3000

Name	Type	Size	Data
COMPANYID	NUMBER	9	List(select companyid from company)
AGENTID	NUMBER	9	List(select agentid from agent)
*			

אחרי שהכניס 3000 רשומות ביותר מ- 3000 ניסיונות בגלל מפתח זר.
השגיאה המתקבלת:





ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

Product

זמן ריצה: 12.078

PRODUCT				
Owner	Table	Number of records		
TAMRAM	PRODUCT	3000..3000		
Name	Type	Size	Data	
PRODUCTBARCODE	NUMBER	10	Sequence(10, [Inc], [WithinParent])	
PRODUCTNAME	CHAR	30	FirstName	
CATEGORYID	NUMBER	9	List(select categoryid from category)	
COMPANYID	NUMBER	9	List(select companyid from company)	
PRODUCTPRICE	FLOAT	22	Components.Price	

OrderFromBasement

זמן ריצה: 13.813

ORDERFROMBASEMENT				
Owner	Table	Number of records		
TAMRAM	ORDERFROMBASEMENT	3000..3000		
Name	Type	Size	Data	
ORDERNUMBER	NUMBER	9	Sequence(10, [Inc], [WithinParent])	
BRANCHID	NUMBER	9	List(select branchid from branch)	
AGENTID	NUMBER	9	List(select agentid from agent)	
ORDERDATE	DATE		Random(1/1/2000, 1/1/2020)	



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

OrderedProduct

זמן ריצה: 24.94

ORDEREDPRODUCT				
Owner	Table	Number of records		
TAMRAM	ORDEREDPRODUCT	3000..3000		
Name	Type	Size	Data	
PRODUCTBARCODE	NUMBER	10	List(select productbarcode from product)	
ORDERNUMBER	NUMBER	9	List(select ordernumber from orderFromBasement)	
ORDERAMOUNT	NUMBER		Random(0, 10000)	
EXPIRYDATE	DATE		Random(1/1/1990, 1/1/2025)	
SOLDAMOUNT	NUMBER		Random(0, 4000)	

Returns

זמן ריצה: 13.875

RETURNS				
Owner	Table	Number of records		
TAMRAM	RETURNS	3000..3000		
Name	Type	Size	Data	
RETURNSREASON	VARCHAR2	500	Components.Description	
RETURNSID	NUMBER	9	Sequence(10, [Inc], [WithinParent])	
RETURNSDATE	DATE		Random(1/1/2000, 1/1/2018)	
PRODUCTBARCODE	NUMBER	10	List(select productBarcode from product)	



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

[Index](#)

Returns

```
create index index_returns  
on returns(returnsDate);  
drop index index_returns;
```

הרצה על שאילתה 1

זמן ריצה אחרי index: 0.046

זמן ריצה ללא Index: 0.076

Ordered Product

```
create index index_orderedProduct  
on orderedproduct(EXPIRYDATE);  
drop index index_orderedProduct;
```

הרצה על שאילתה 4

זמן ריצה אחרי index: 0.060

זמן ריצה ללא Index: 0.105

Ordered Product

```
create index index_orderedProduct_best  
on orderedproduct(orderAmount);  
drop index index_orderedProduct_best;
```

הרצה על שאילתה 6

זמן ריצה אחרי index: 0.034

זמן ריצה ללא Index: 0.041



ניהול מחסן

בס"ד

רחלי אסליזודה ותהילה עמורם

Ordered Product

```
create index index_left  
on orderedproduct (orderAmount, soldamount);  
drop index index_left;
```

הרצה על שאילתה 5

זמן ריצה אחרי index: 0.086

זמן ריצה ללא Index: 0.096



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

[Text importerter](#)

נעשה לטבלה category

להלן קובץ הטקסט אותו נייבא:

```
Z:\osherAd\textImp - Notepad++
File Edit Search View Encoding Lang
[Icons]
textImp x
1 4000, 'snacks'
2 4001, 'nuts'
3 4002, 'drinks'
4
5
```

ההגדרות של הייבוא:

Data from Textfile Data to Oracle

File Data Encoding

Configuration

General

Fieldcount: 2

☒ End at line-end

☐ Name in header

☒ Skip empty lines

Quote character: "

Comment line:

Import lines: 1 ..

Field1 (+0 .. +0)

Field2 (+0 .. +0)

Field Start

☐ Relative position

☐ Absolute position

☐ Character

Field End

☐ Length

☐ Character

Filter: Apply

Result Preview



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

Data from Textfile Data to Oracle

General

Owner: TAMRAM Table: CATEGORY

Commit every...: 100

☐ Overwrite duplicates ☐ Delete records
☒ Ignore duplicates ☐ Truncate table

Initializing Script: ...
Finalizing Script: ...

Fields

Field1 -> CATEGORYID
Field2 -> CATEGORYNAME

Field: CATEGORYID (NUMBER) Fieldtype: Number

Create SQL

SQL function: ...
additional Oracle processing, for example: substr(#, 1, 20)

Data from Textfile Data to Oracle

General

Owner: TAMRAM Table: CATEGORY

Commit every...: 100

☐ Overwrite duplicates ☐ Delete records
☒ Ignore duplicates ☐ Truncate table

Initializing Script: ...
Finalizing Script: ...

Fields

Field1 -> CATEGORYID
Field2 -> CATEGORYNAME

Field: CATEGORYNAME (VARCHAR2) Fieldtype: String

Create SQL

SQL function: ...
additional Oracle processing, for example: substr(#, 1, 20)

Result Preview



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

Data from Textfile | Data to Oracle

General

Owner: TAMRAM | Table: CATEGORY

Commit every...: 100

☐ Overwrite duplicates ☐ Delete records

☒ Ignore duplicates ☐ Truncate table

Initializing Script: ...

Finalizing Script: ...

Fields

Field1 -> CATEGORYID
Field2 -> CATEGORYNAME

Field: CATEGORYNAME (VAR) | Fieldtype: String

Create SQL

SQL function: ...
additional Oracle processing, for example: substr(#, 1, 20)

Result Preview

1	2
4001	'nuts'
4002	'drinks'

Import | Import to Script | Close | tamram@labdbwin | 3 records imported | Help

נעשה יבוא לקובץ, נקרא לו txtimp וקיבלנו את הקובץ הבא:

representDG.gd | productInsert.sql | Edit data CATEGORY@LABDBWIN | txtImp | txtimp.sql

SQL | Output | Statistics

```
insert into tamram.category (categoryid, categoryname)
values (4000, ''snacks'');

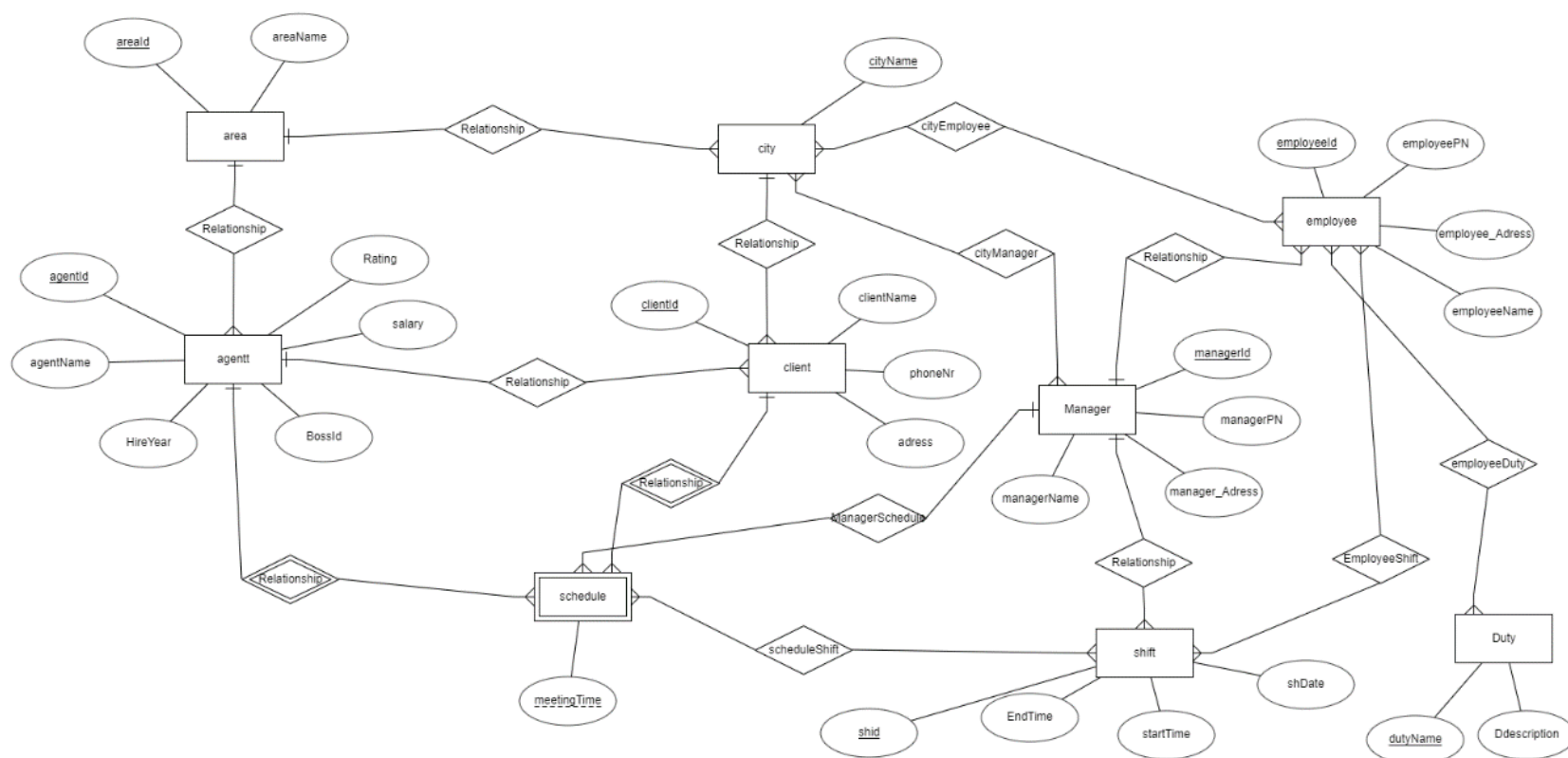
insert into tamram.category (categoryid, categoryname)
values (4001, ''nuts'');

insert into tamram.category (categoryid, categoryname)
values (4002, ''drinks'');

commit;
```

רחלי אסליזדה ותהילה עמרם

ERD מורחב לפני אינטגרציה:





ניהול מחסן

בס"ד

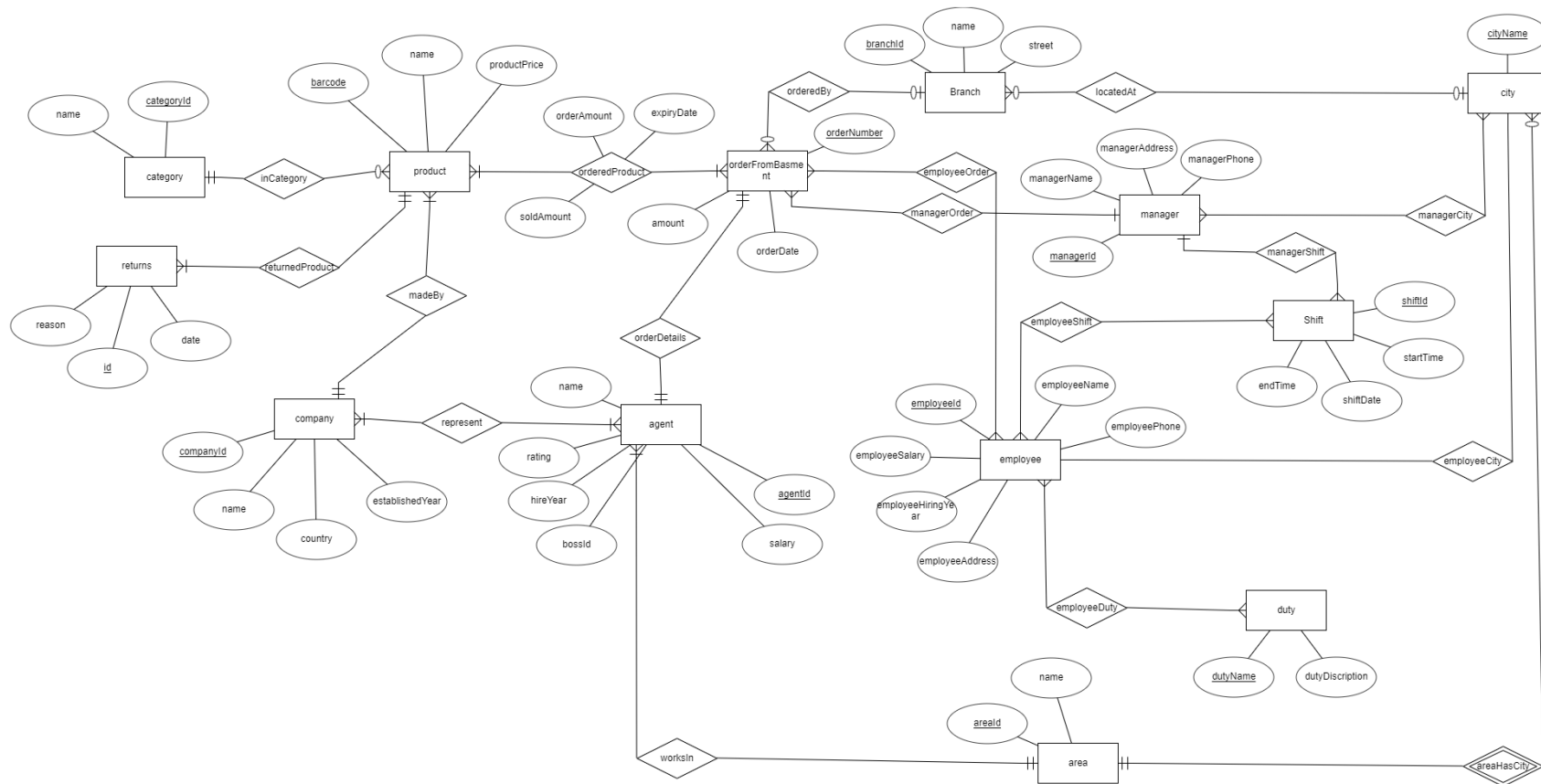
רחלי אסליזודה ותהילה עמורם



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם
ERD מורחב אחרי אינטגרציה:





ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

יצירה ואכלוס טבלאות - האינטגרציה:

```
-- create
CREATE TABLE Manager
(
    managerId INT NOT NULL,
    managerPN varchar2(20) NOT NULL,
    manager_Adress varchar2(20) NOT NULL,
    managerName varchar2(20) NOT NULL,
    PRIMARY KEY (managerId)
);

CREATE TABLE shift
(
    shid INT NOT NULL,
    shDate date NOT NULL,
    startTime INT NOT NULL,
    EndTime INT NOT NULL,
    managerId INT NOT NULL,
    PRIMARY KEY (shid),
    FOREIGN KEY (managerId) REFERENCES Manager(managerId)
);

CREATE TABLE Duty
(
    dutyName varchar2(20) NOT NULL,
    Ddescription varchar2(50) NOT NULL,
    PRIMARY KEY (dutyName)
);

CREATE TABLE employee
(
    employeeId INT NOT NULL,
    employeePN number NOT NULL,
    employee_Adress varchar2(20) NOT NULL,
    employeeName varchar2(20) NOT NULL,
    employeeSalary INT NOT NULL,
    hiringYear INT NOT NULL,
    managerId INT NOT NULL,
    PRIMARY KEY (employeeId),
    FOREIGN KEY (managerId) REFERENCES Manager(managerId)
);
```



רחלי אסליזדה ותהילה עמרם

```
CREATE TABLE cityManager
(
    cityName varchar2(20) NOT NULL,
    managerId INT NOT NULL,
    PRIMARY KEY (cityName, managerId),
    FOREIGN KEY (cityName) REFERENCES city(cityName),
    FOREIGN KEY (managerId) REFERENCES Manager(managerId)
);
CREATE TABLE EmployeeShift
(
    employeeId INT NOT NULL,
    shid INT NOT NULL,
    PRIMARY KEY (employeeId, shid),
    FOREIGN KEY (employeeId) REFERENCES employee(employeeId),
    FOREIGN KEY (shid) REFERENCES shift(shid)
);
CREATE TABLE employeeDuty
(
    employeeId INT NOT NULL,
    dutyName varchar2(20) NOT NULL,
    PRIMARY KEY (employeeId, dutyName),
    FOREIGN KEY (employeeId) REFERENCES employee(employeeId),
    FOREIGN KEY (dutyName) REFERENCES Duty(dutyName)
);
CREATE TABLE cityEmployee
(
    cityName varchar2(20) NOT NULL,
    employeeId INT NOT NULL,
    PRIMARY KEY (cityName, employeeId),
    FOREIGN KEY (cityName) REFERENCES city(cityName),
    FOREIGN KEY (employeeId) REFERENCES employee(employeeId)
);
CREATE TABLE employeeOrder
(
    employeeId INT NOT NULL,
    orderNumber INT NOT NULL,
    PRIMARY KEY (employeeId, orderNumber),
    FOREIGN KEY (employeeId) REFERENCES employee(employeeId),
    FOREIGN KEY (orderNumber) REFERENCES orderFromBasement(orderNumber)
);
```

המשך יצירה ואכלוס טבלאות - שלב האינטגרציה:



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

```
-- Add/modify columns  
alter table ORDERFROMBASEMENT add managerid integer;
```

```
-- drop  
drop table schedule;  
drop table client;
```



ניהול מחסן

בס"ד

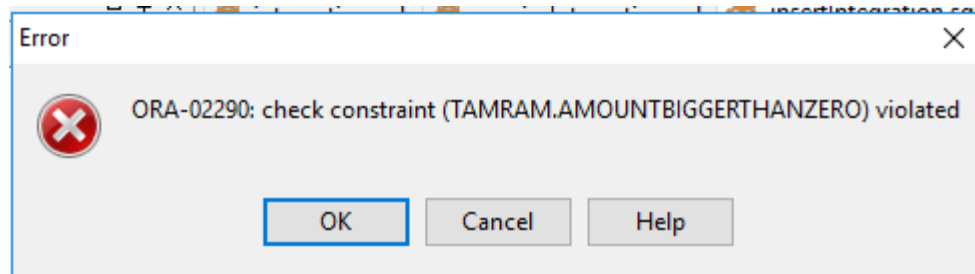
רחלי אסליזדה ותהילה עמרם

אילוצים

1. מוודא שהכמות המוזמנת ממוצר גדולה מ-0

```
-- check 1
-- Create/Recreate check constraints
alter table ORDEREDPRODUCT
add constraint amountBiggerThanZero
check (orderAmount > 0);
```

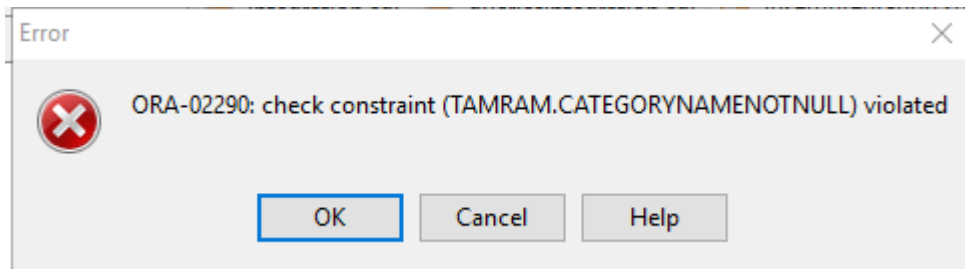
2.
וודא
האם
שם
קטגורי
ת



המוצר אינו ריק.

```
-- check 2
-- Create/Recreate check constraints
alter table CATEGORY
add constraint CategoryNameNotNull
check (categoryName is not null);
```

3.
וודא
שזמן
התחלת



המשמרת הוא לפני זמן סיום המשמרת

```
-- check 3
-- Create/Recreate check constraints
alter table SHIFT
add constraint startShiftBeforeEndShift
check (startTime < endTime);
. . .
```



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

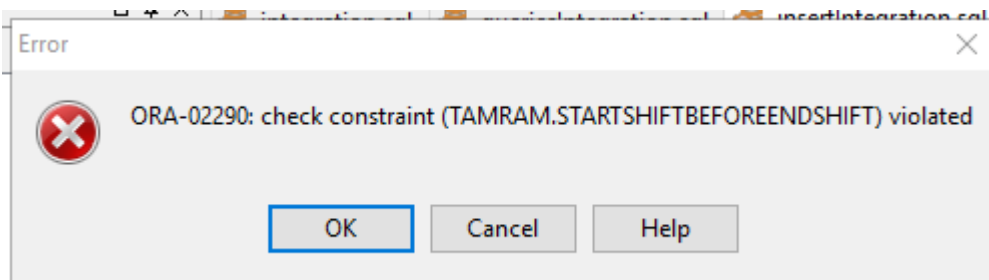
```
-- drop constraint  
alter table SHIFT drop constraint min3HoursPerShift;
```

מאחר ויש לנו 2 אילוצים חופפים זמן התחלת משמרת קטן מזמן סיום משמרת וכן משמרת היא לפחות 3 שעות.

ע"מ לקבל הודעת שגיאה בגלל האילוץ הנוכחי עשינו drop לאילוץ החופף

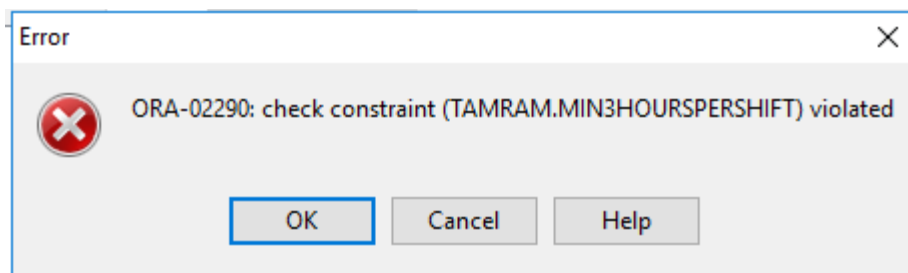
4.

וודא
שאורך
המשמר
ת הוא
מינימום
שלוש



שעות

```
-- check4  
-- Create/Recreate check constraints  
alter table SHIFT  
add constraint min3HoursPerShift  
check (endTime - startTime >=3);|
```





ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

[:View1](#)

ניצור מבט על הטבלאות בשם employeeShiftDetails המבט מיועד למנהל

```
--view1
CREATE or replace VIEW employeeShiftDetails
(employeeId, shid, employeeName, employeePN, employee_Address, employeeSalary, hiringYear, ManagerId, startTime, EndTime, shDate)
AS
SELECT employeeId, shid, employeeName, employeePN, employee_Address, employeeSalary, hiringYear, ManagerId, startTime, EndTime, shDate
FROM shift natural join employee natural join EmployeeShift ;
```

משמרת בו הוא יכול לראות את הפרטים של העובדים במשמרות.

[שאילתה 1 על ה-View](#)

```
--query view1
SELECT employeeId as id, employeeName as name, employeeSalary as salary, startTime as "start", EndTime as "end"
FROM employeeShiftDetails
where shid = &<name="shift" list="select shid from shift" description="no" restricted="yes">
```

בחירת עובד והצגת פרטי העובדים שעבדו באותה משמרת שנבחרה.

	ID	NAME	SALARY	start	end
▶	1	15801096	Aviel Levy ...	8000	8 13

[מחיקת :View1](#)

```
--drop view 1
drop view employeeShiftDetails;
```



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

[:View2](#)

ניצור מבט על הטבלאות בשם orderProductDetails המבט מיועד לאחראי ההזמנות של הסניף ובו הוא יכול לראות את היסטורית ההזמנות וכן את מצב המלאי.

```
-- view2
create or replace view ordersProductDetails
(orderNumber,branchId,agentId,orderDate,productBarcode, productName, categoryId, productPrice,companyId,orderAmount,expiryDate, soldAmount)
as select orderNumber,branchId,agentId,orderDate,productBarcode, productName, categoryId, productPrice,companyId,orderAmount,expiryDate, soldAmount
from orderfrombasement natural join product natural join orderedproduct
```

[שאלתה 2 על ה-View](#)

בחירת תאריך ומביא את כל ההזמנות שהוזמנו באותו תאריך

וגם שהכמות שהוזמנה שווה לכמות שנמכרה.

```
SELECT *
FROM ordersProductDetails
where orderDate = to_date(&<name=d_from type="string">,'DD/MM/YYYY') and
orderamount=soldamount
```

ORDERNUMBER	BRANCHID	AGENTID	ORDERDATE	PRODUCTBARCODE	PRODUCTNAME	CATEGORYID	PRODUCTPRICE	COMPANYID	ORDERAMOUNT	EXPIRYDATE	SOLDAMOUNT		
1	4	4	615988674	20/06/2018	2	soy sausages	...	1	20	4	500	13/09/2010	500

[מחיקת View2](#)

```
--drop view 2
drop view ordersProductDetails;
```



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

דוחות:

דו"ח 1:

היסטוריה של הזמנת מוצרים לפי סניף וחודש.

המשתמש יבחר את הסניף הרצוי מתוך רשימה ויכניס את החודש ויקבל בדוח את היסטוריית ההזמנות שהתבצעו בחודש שהזין.

```
select branchname as "branch",orderdate as "date",ordernumber as "order number",
productbarcode as "barcode", productname as "name", categoryname as "category",
companyname as "company", productprice as "price",orderamount as "amount" ,
productprice * orderamount as "sum"
from product natural join orderedproduct natural join orderfrombasement natural join
company natural join category natural join branch
where branchid = &<name="branch" list="select branchid, branchname
from branch order by branchname" description="yes" restricted="yes">
and EXTRACT(month FROM orderdate) = &month
order by ordernumber;
```

Name	Value
branch	Nazareth Illit hagadol
month	6

OK Cancel Clear



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

branch = 4						
month = 6						
branch Nazareth Ilit hagadol						
date 20/06/2018						
order number 4						
barcode	name	category	company	price	amount	sum
2	soy sausages	frozen	Hanasich	20	500	10000
5	tomato	vegetables	Pilot	8	6	48
3	bleach	cleanliness	Osem	34	90	3060
						13108
branch Nazareth Ilit hagadol						
date 19/06/2019						
order number 8						
barcode	name	category	company	price	amount	sum
2	soy sausages	frozen	Hanasich	20	70	1400
5	tomato	vegetables	Pilot	8	98	784
						2184
branch Nazareth Ilit hagadol						
date 14/06/2010						
order number 91						
barcode	name	category	company	price	amount	sum
5	tomato	vegetables	Pilot	8	6	48
1	corn	canned food	Taaman	40	100	4000
2	soy sausages	frozen	Hanasich	20	500	10000
3	bleach	cleanliness	Osem	34	90	3060
						17108
branch Nazareth Ilit hagadol						
date 23/06/2014						
order number 92						
barcode	name	category	company	price	amount	sum
3	bleach	cleanliness	Osem	34	70	2380
5	tomato	vegetables	Pilot	8	98	784
2	soy sausages	frozen	Hanasich	20	98	1960
						5124



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם

דו"ח 2:

היסטוריה של הזמנות מוצרים כללית לפי חודשים.

```
select EXTRACT(month FROM orderdate) as "month", productname as "name",  
productbarcode as "barcode", orderamount as "amount", productprice * orderamount as "price"  
from product natural join orderedproduct natural join orderfrombasement  
order by productbarcode;
```

month 6			
name	barcode	amount	price
corn	1	100	4000
corn	1	100	4000
soy sausages	2	98	1960
soy sausages	2	70	1400
soy sausages	2	500	10000
		868	21360
month 5			
name	barcode	amount	price
soy sausages	2	98	1960
		98	1960
month 6			
name	barcode	amount	price
soy sausages	2	500	10000
bleach	3	90	3060
bleach	3	90	3060
bleach	3	70	2380
bleach	3	380	12920
flour	4	786	1752.78
tomato	5	98	784
tomato	5	6	48
tomato	5	6	48
tomato	5	98	784
tomato	5	98	784
		2222	35620.78



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם



ניהול מחסן

בס"ד

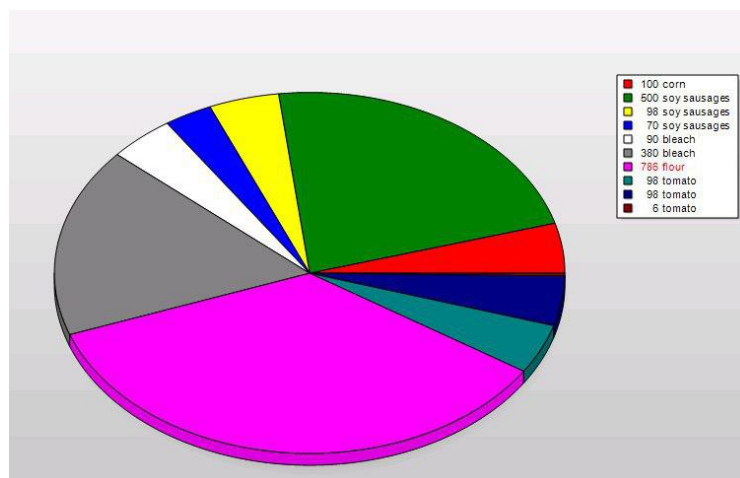
רחלי אסליזדה ותהילה עמורם

גרפים:

גרף 1:

אחוז מכירת מוצר מתוך טווח המחירים

```
select orderamount, productname  
from product natural join orderedproduct  
.
```





ניהול מחסן

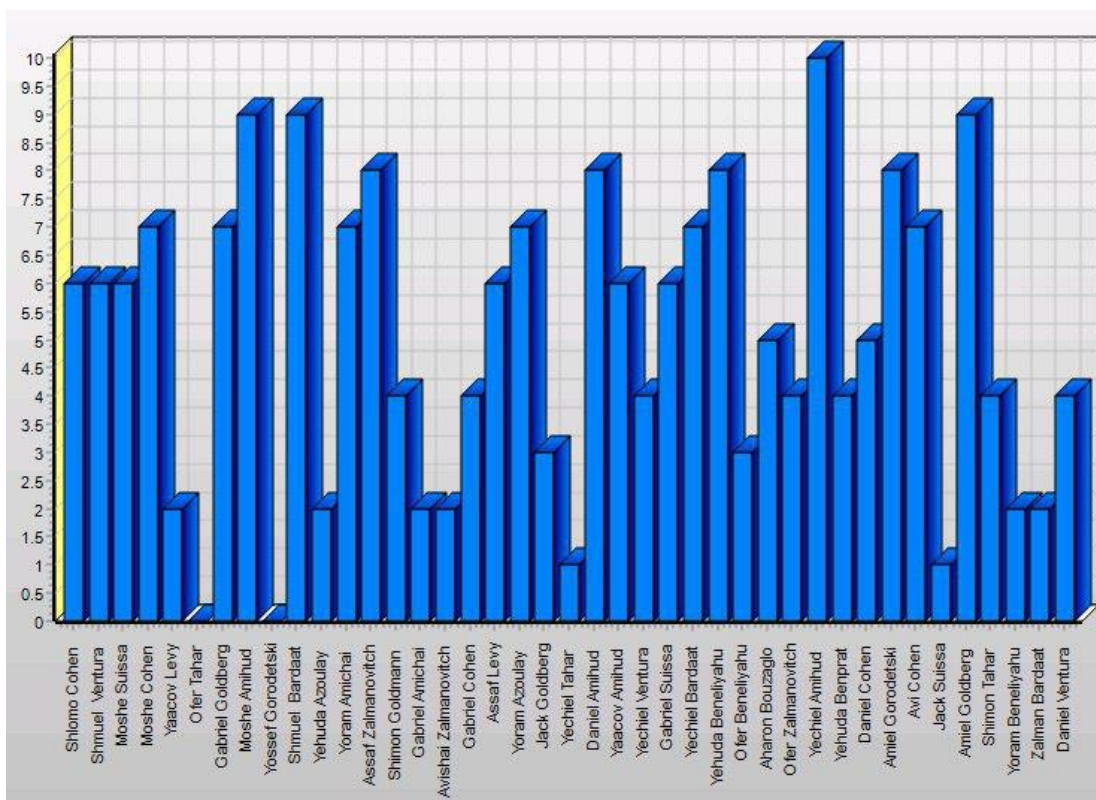
בס"ד

רחלי אסליזדה ותהילה עמרם

גרף 2:

הצגת דירוג סוכן ביחד לשאר

```
select agentrating, agentname
from agent
```





ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם
הרשאות גישה:

insert: נתינת הרשאה של

```
-- permission 1  
GRANT INSERT ON Category TO aslizada;  
REVOKE INSERT ON Category FROM aslizada;
```

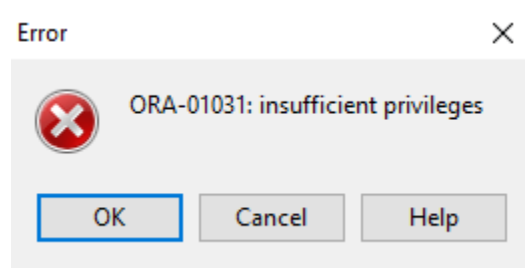
שימוש בהרשאת גישה:

```
INSERT INTO TAMRAM.category (categoryid,categoryname)  
VALUES (008,'yyy');  
commit;
```

הטבלה לאחר השימוש בהרשאה:

	CATEGORYID	CATEGORYNAME
1	8	yyy
2	1	frozen
3	2	vagetables
4	3	baking
5	4	cleanliness

ניסיון הכנסה אחרי שלילת ההרשאה ע"י revoke:





ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם
[המשך הרשאות:](#)

[נתינת הרשאה של select:](#)

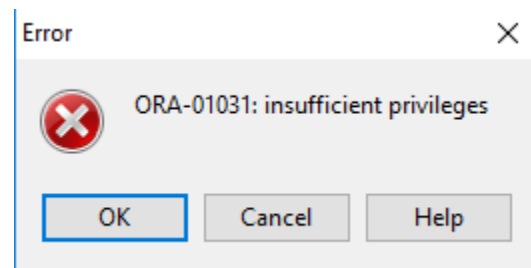
```
-- permission 2  
GRANT SELECT ON Product TO aslizada;  
REVOKE SELECT ON Product FROM aslizada;
```

שימוש בהרשאה והטבלה המתקבלת:

```
select * from tamram.product
```

	PRODUCTBARCODE	PRODUCTNAME	CATEGORYID	COMPANYID	PRODUCTPRICE
1	1	corn	5	5	40
2	2	soy sausages	1	4	20
3	3	bleach	4	3	34
4	4	flour	3	2	2.23
5	5	tomato	2	1	8
6	6	cucamber	2	1	8

ניסיון הכנסה אחרי שלילת ההרשאה ע"י revoke:





ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

פרוצדורות:

1. הפרוצדורה מקבלת מספר הזמנה.

לפי מספר ההזמנה היא מוצאת את תאריך ההזמנה.

לאחר מכן, היא מוצאת את כל המשמרות באותו תאריך, ובחרת את המשמרת הראשונה שמספר העובדים שלה לא ריק (ז"א שקיים לפחות עובד 1) ומשייכת את כל עובדי המשמרת להזמנה.

```
INSERT INTO orderfrombasement (orderNumber,branchId,agentId,orderDate,managerId)
VALUES (30,5,564441324,to_date('25/06/2019', 'DD/MM/YYYY'),158015987);
--
INSERT INTO shift (shid,startTime,EndTime, shDate,ManagerId)
VALUES (30,8,13, to_date('25/06/2019', 'DD/MM/YYYY HH24:MI'),158010866);
--
INSERT INTO EmployeeShift (employeeId, shid)
VALUES (15801096,30);
--
INSERT INTO EmployeeShift (employeeId, shid)
VALUES (564841324,30);
--
INSERT INTO EmployeeShift (employeeId, shid)
VALUES (560458763,30);
```

```
--order_date orderfrombasement.orderdate$type;
create or replace procedure connectEmployeeToOrder(oNumber number) is
CURSOR c_employee is
    SELECT employeeid,shid FROM employeeshift ;
    order_date orderfrombasement.orderdate$type;
    shift_id shift.shid$type;
    shift_id_e shift.shid$type;
    employee_id employeeshift.employeeid$type;
begin
    -- order date
    select orderDate
    into order_date
    from orderfrombasement
    where orderfrombasement.ordernumber = oNumber;
    -- select first shift in the same date with at least 1 employee
    select s.shid
    into shift_id
    from shift s, (SELECT count(shid) AS total_amt,shid
    FROM employeeshift
    GROUP BY shid) query1
    where s.shdate = order_date and query1.total_amt>0 and rownum = 1;
    -- all the employee in the selected shift
    OPEN c_employee;
    LOOP
        FETCH c_employee into employee_id,shift_id_e;
        EXIT WHEN c_employee%notfound;
        if(shift_id = shift_id_e)then
            dbms_output.put_line(employee_id|| ' ' || oNumber);
            insert into EMPLOYEEORDER (EMPLOYEEID, ORDERNUMBER)
            values (employee_id, oNumber);
            END IF;
        END LOOP;
    CLOSE c_employee;
end connectEmployeeToOrder;
```

	EMPLOYEEID	ORDERNUMBER
1	15801096	30
2	560458763	30
3	564841324	30



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמורם
2.מחיקת עובדים מהזמנה

(מקבלת מספר הזמנה ומוחקת את כל העובדים שממונים על אותה הזמנה.)
למשל: במקרה שההזמנה תגיע באיחור או שההזמנה התבטלה.

```
create or replace procedure deleteEmployeeFromOrder(oNumber number) is
cursor c_employee_order is
select *
from employeeorder ;
rec_employee_order employeeorder%rowtype;
order_number employeeorder.ordernumber%type;
begin
open c_employee_order;
loop
fetch c_employee_order into rec_employee_order;
exit when c_employee_order%notfound;
if(rec_employee_order.ordernumber = oNumber)then
delete
from employeeorder
where rec_employee_order.ordernumber = oNumber;
end if;
end loop;
close c_employee_order;
end deleteEmployeeFromOrder;
```



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

פונקציות:

1. הפונקציה מחזירה את שם החברה שהכי רווחית.
מכפילה עבור כל מוצר (מחיר כפול כמות) ואז עושה סכום של כל המוצרים ובודקת של איזו חברה הסכום היה הכי גבוה ומחזירה את שם החברה הזו.

```
create or replace function getMaxSellsCompany return varchar2 is
FunctionResult varchar2(30);
CURSOR c_company is
    SELECT companyid,companyname FROM company ;
CURSOR c_orderedProduct is
    SELECT orderamount,productprice,companyid FROM product natural join orderedproduct;
product_barcode product.productbarcode%type;
product_amount orderedproduct.orderamount%type;
product_price product.productprice%type;
ccompany_id company.companyid%type;
pcompany_id product.companyid%type;
company_name company.companyname%type;
current_sum number;
max_sum number;
current_company varchar2(30);
begin

current_sum:=0;
max_sum:=0;
current_company:= 'no company was found';
OPEN c_company;
LOOP
    FETCH c_company into ccompany_id,company_name;
    EXIT WHEN c_company%notfound;
    current_sum:=0;
    OPEN c_orderedProduct;
    FETCH c_orderedProduct into pcompany_id,product_price,product_amount;
    EXIT WHEN c_orderedProduct%notfound;
    current_sum:= current_sum+product_price*product_amount;
    CLOSE c_orderedProduct;
    if (current_sum >= max_sum) then
        max_sum := current_sum;
        current_company:=company_name;
    END IF;
END LOOP;
CLOSE c_company;

FunctionResult:= current_company;
return(FunctionResult);
end getMaxSellsCompany;
```

Variable	Type	Value
result	String	Osem



ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

2. סופרת כמה עובדים מצטיינים יש

(שסכום השעות שעבד עבור חודש מסוים הוא הגבוה ביותר) ומציגה את התוצאה שלהם על המסך.

```
create or replace function bonusTheBestMonthlyEmployee return number is
FunctionResult number;
CURSOR c_employee is
    SELECT employeesalary, employeeid FROM employee;
CURSOR c_shift is
    SELECT starttime,endtime,shdate,employeeid FROM shift natural join employeesshift;
current_sum number;
max_sum number;
counter number;
one_number integer;
begin
    one_number := 1;
    current_sum:=0;
    max_sum:=0;
    counter:=0;
FOR i IN c_employee LOOP
current_sum := 0;

    for j in c_shift loop

        if(j.employeeid= i.employeeid and extract(year from j.shdate) = EXTRACT(year from sysdate) and
EXTRACT(month from sysdate) = EXTRACT(month from j.shdate)) then
current_sum:= current_sum + (j.endtime-j.starttime);
            end if;

    end loop;
    if (current_sum >= max_sum) then
        dbms_output.put_line(counter);
        dbms_output.put_line(i.employeeid);
        max_sum:= current_sum;
        counter:=counter+1;
    end if;
END LOOP;
    FunctionResult:=counter;
    return (FunctionResult);
end bonusTheBestMonthlyEmployee;
```

Variable	Type	Value
result	Float	3



ניהול מחסן

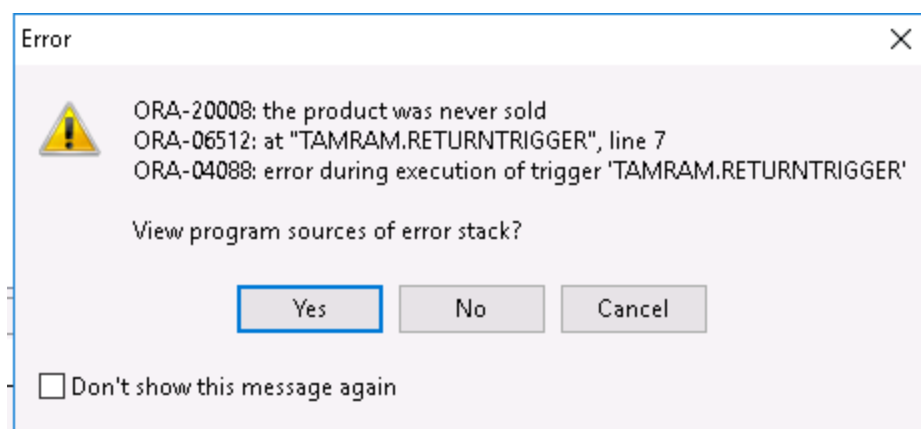
בס"ד

רחלי אסליזדה ותהילה עמרם

Trigger:

1. למנוע החזרת מוצר שלעולם לא נרכש.

```
create or replace trigger returnTrigger
before insert
on returns
-- didn't mark statements
for each row
declare
cursor ordered_product is select * from orderedproduct;
-- local variables here
begin
for i in ordered_product loop
if i.productbarcode != :new.productbarcode then
raise_application_error(-20008,'the product was never sold');
end if;
end loop;
end returnTrigger;
```





ניהול מחסן

בס"ד

רחלי אסליזדה ותהילה עמרם

2. אם משנים מחיר מוצר לאפס, שלא יאפשר זאת ויקפיץ שגיאה.

```
create or replace trigger updateProductPriceToZero
before update
on product
for each row
declare
-- local variables here
begin
if :new.productprice= 0 then
raise_application_error(-20009,'product price must be >0');
end if;
end updateProductPriceToZero;
```

ביצוע עדכון:

וכתוצאה מכך חריגה

```
update product set product.productprice = 0
where productbarcode = 9;
```

