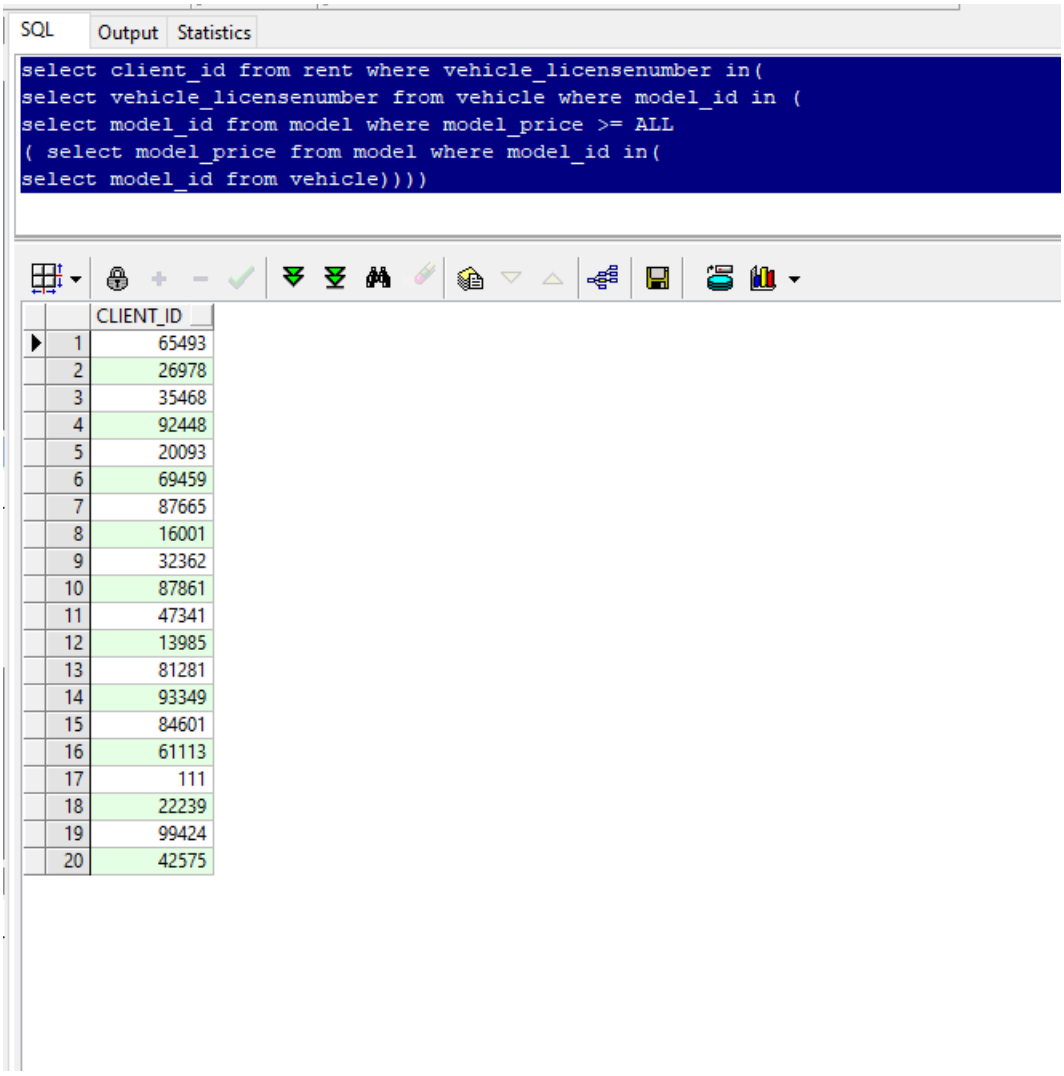
דו"ח פרויקט טיולים השכרת רכב שלבים 2 3 4:

שאילתות שלב 4:

שאילתה הבוחרת את כל הלקוחות ששילמו את המחיר הגבוה ביותר על הרכב שהם שכרו:

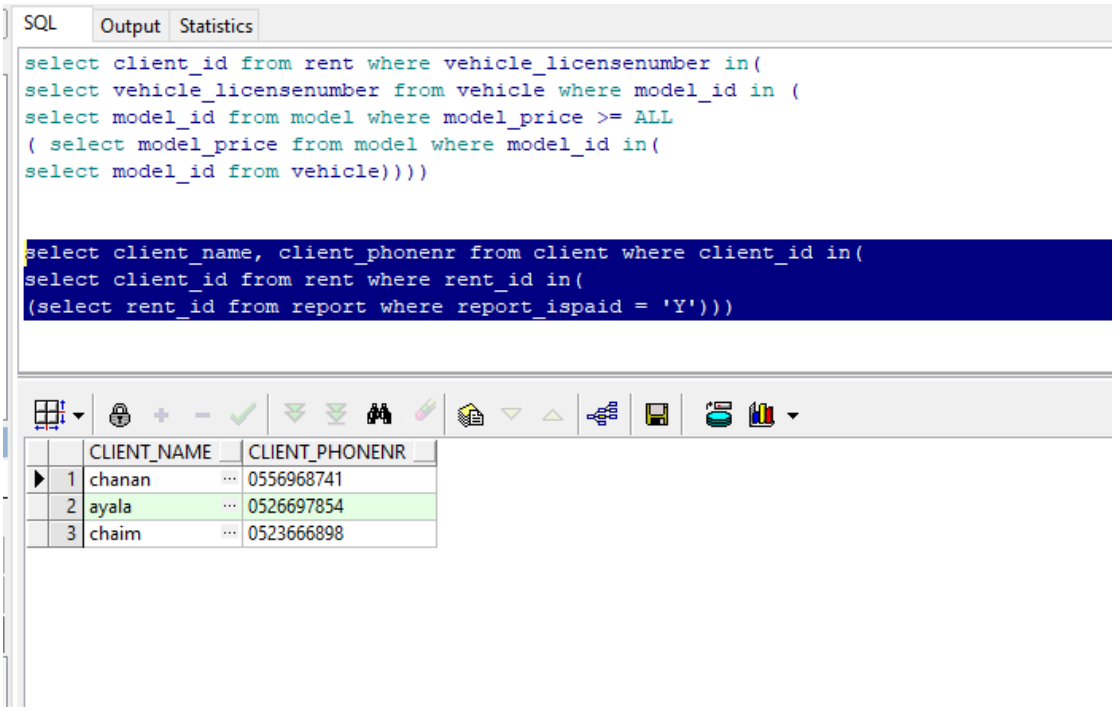
```
select client_id from rent where vehicle_licensenumber in(
select vehicle_licensenumber from vehicle where model_id in (
select model_id from model where model_price >= ALL
( select model_price from model where model_id in(
select model_id from vehicle))))
```



	CLIENT_ID
1	65493
2	26978
3	35468
4	92448
5	20093
6	69459
7	87665
8	16001
9	32362
10	87861
11	47341
12	13985
13	81281
14	93349
15	84601
16	61113
17	111
18	22239
19	99424
20	42575

שאילתה הבוחרת רק את הלקוחות שביצעו השכרה וששילמו עליה:

```
select client_name, client_phonenr from client where client_id in(
select client_id from rent where rent_id in(
(select rent_id from report where report_ispaid = 'Y')))
```



שאילתה הבוחרת רק את הסוכנים שעשו יותר מהשכרה אחת:

```
select agent_name from agent
where agent_id in(
select agent_id from rent
group by agent_id
```

```
having count(*) > 1)
```

SQL Output Statistics

```
select model_id from model where model_price >= ALL
( select model_price from model where model_id in(
select model_id from vehicle))))


select client_name, client_phonenr from client where client_id in(
select client_id from rent where rent_id in(
(select rent_id from report where report_ispaid = 'Y')))


select agent_name from agent
where agent_id in(
select agent_id from rent
group by agent_id
having count(*) > 1)


select a.agent_id, c.client_id from
agent_rating a, client c, (select agent_id, client_id from rent) r
where a.agent_id = r.agent_id and c.client_id = r.client_id and a.city_id = c.c

update agent_rating
```

	AGENT_NAME
1	Lee
2	Alex
3	Pelvic
4	Will
5	Annie
6	Cole
7	Adrien
8	Marina
9	Orlando
10	Daniel
11	James
12	Lee
13	Morgan
14	Ray
15	yaakov
16	Janice
17	Albert
18	Julianne
19	Beverley
20	Jeroen
21	Cameron

& 13:1 0:01 SYSTEM@XE 21 rows selected in 1.766 seconds (more...)

השאילתה הבאה בוחרת את כל התז של סוכן ולקוח שהיו באותה עסקה וגם גרים באותה עיר:

```
select a.agent_id, c.client_id from
agent a, client c, (select agent_id, client_id from rent) r
```

```
where a.agent_id = r.agent_id and c.client_id = r.client_id and
a.city_id = c.city_id
```

SQL Output Statistics

```
select agent_name from agent
where agent_id in(
select agent_id from rent
group by agent_id
having count(*) > 1)


select a.agent_id, c.client_id from
agent a, client c, (select agent_id, client_id from rent) r
where a.agent_id = r.agent_id and c.client_id = r.client_id and a.city_id = c.city_id

update agent_rating
set agent_salary = agent_salary + 1000 where agent_id in(
select agent_id from rent
group by agent_id
having count(*) > 10) and agent_salary < 5000

update agent_rating
set report_ispaid = 'Y' where client_id in(
select client_id from rent where rent_id in(
```

	AGENT_ID	CLIENT_ID
1	32374	99297
2	7	88157
3	6	88792
4	91753	52432
5	29621	34618
6	32374	73324
7	6	52852
8	99835	94974
9	96931	20488
10	6	111
11	25287	15723
12	65682	15631
13	98599	78797
14	82338	62896
15	32932	97544
16	65285	73396
17	83871	54244
18	17339	19557
19	49477	13985
20	6	36852
21	49477	43835

20:1 SYSTEM@XE 21 rows selected in 0.047 seconds (more...)

שאילתות עדכון:

נרצה להוסיף 1000 שח למשכורות של כל הסוכנים שביצעו יותר מ7 השכרות והמשכורת שלהם נמוכה מ15000:

השאילתה:

```
update agent
set agent_salary = agent_salary + 1000 where agent_id in(
select agent_id from rent
group by agent_id
having count(*) > 7) and agent_salary < 15000
```

לפני העדכון:

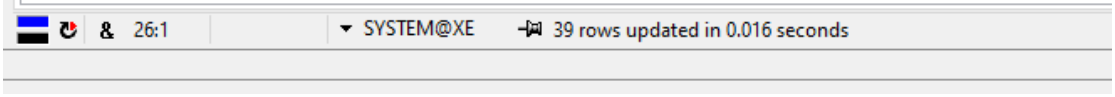
SQL Output Statistics

```
where agent_id in(
select agent_id from rent
group by agent_id
having count(*) > 1)


select a.agent_id, c.client_id from
agent a, client c, (select agent_id, client_id from rent) r
where a.agent_id = r.agent_id and c.client_id = r.client_id and a.city_id = c.city_id

select * from agent

update agent
set agent_salary = agent_salary + 1000 where agent_id in(
select agent_id from rent
group by agent_id
having count(*) > 7) and agent_salary < 15000

update agent
set report_ispaid = 'Y' where client_id in(
select client_id from rent where rent_id in(
(select rent_id from report where report_ispaid = 'N')))
and agent_name = 'meir'
```

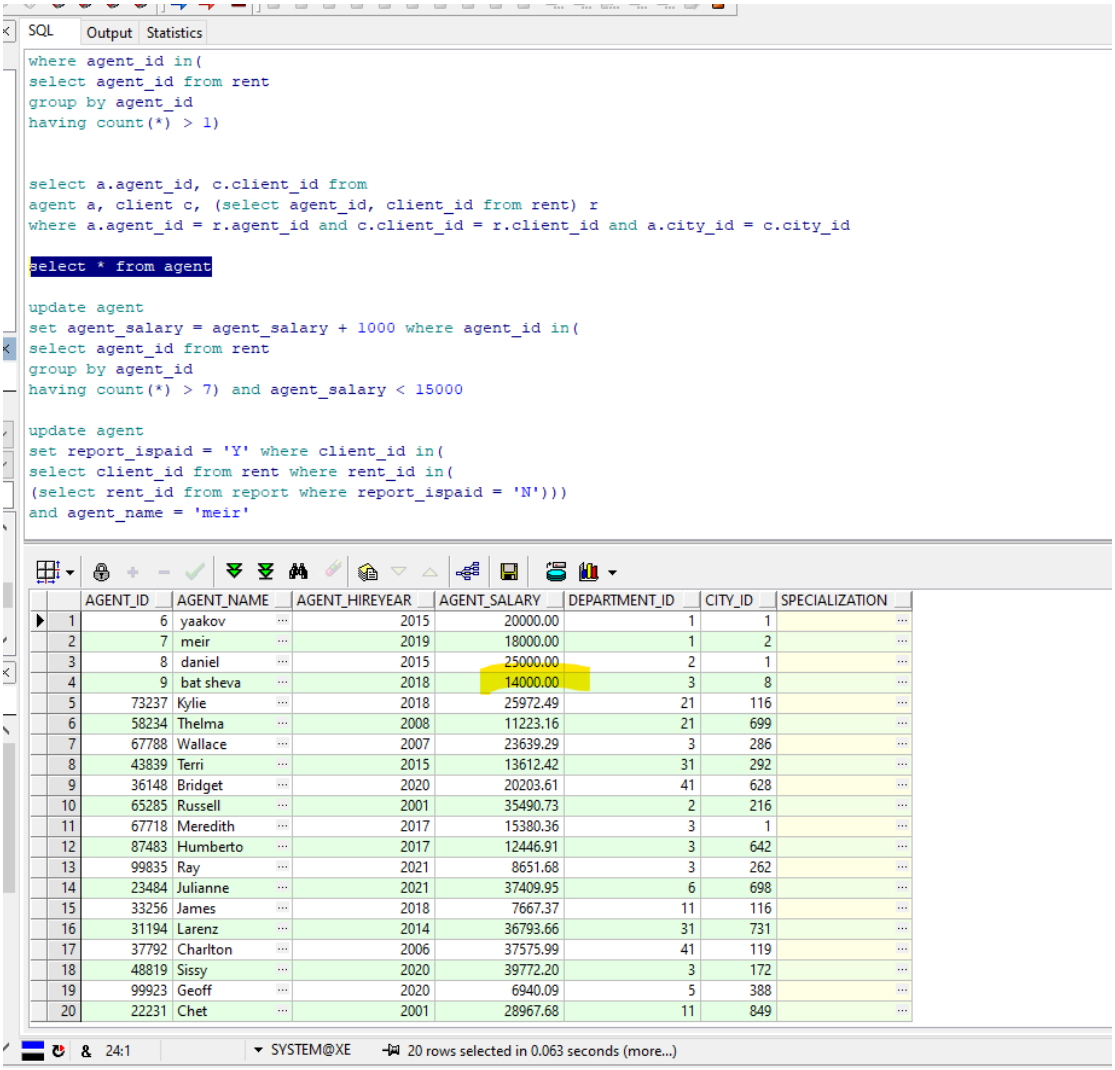
	AGENT_ID	AGENT_NAME	AGENT_HIREYEAR	AGENT_SALARY	DEPARTMENT_ID	CITY_ID	SPECIALIZATION
1	6	yaakov	2015	20000.00	1	1	
2	7	meir	2019	18000.00	1	2	
3	8	daniel	2015	25000.00	2	1	
4	9	bat sheva	2018	13000.00	3	8	
5	73237	Kylie	2018	25972.49	21	116	
6	58234	Thelma	2008	10223.16	21	699	
7	67788	Wallace	2007	23639.29	3	286	
8	43839	Terri	2015	12612.42	31	292	
9	36148	Bridget	2020	20203.61	41	628	
10	65285	Russell	2001	35490.73	2	216	
11	67718	Meredith	2017	14380.36	3	1	
12	87483	Humberto	2017	11446.91	3	642	
13	99835	Ray	2021	7651.68	3	262	
14	23484	Julianne	2021	37409.95	6	698	
15	33256	James	2018	6667.37	11	116	
16	31194	Larenz	2014	36793.66	31	731	
17	37792	Charlton	2006	37575.99	41	119	
18	48819	Sissy	2020	39772.20	3	172	
19	99923	Geoff	2020	5940.09	5	388	
20	22231	Chet	2001	28967.68	11	849	

24:1 SYSTEM@XE 20 rows selected in 0.078 seconds (more...)

```
update agent
set agent_salary = agent_salary + 1000 where agent_id in(
select agent_id from rent
group by agent_id
having count(*) > 7) and agent_salary < 15000

update agent
set report_ispaid = 'Y' where client_id in(
select client_id from rent where rent_id in(
(select rent_id from report where report_ispaid = 'N')))
and agent_name = 'meir'
```



ואכן, לאחר העדכון נראה שלבת שבע למשל נוספו 1000 שח למשכורת:



```
SQL  Output  Statistics
where agent_id in(
select agent_id from rent
group by agent_id
having count(*) > 1)


select a.agent_id, c.client_id from
agent a, client c, (select agent_id, client_id from rent) r
where a.agent_id = r.agent_id and c.client_id = r.client_id and a.city_id = c.city_id

select * from agent

update agent
set agent_salary = agent_salary + 1000 where agent_id in(
select agent_id from rent
group by agent_id
having count(*) > 7) and agent_salary < 15000

update agent
set report_ispaid = 'Y' where client_id in(
select client_id from rent where rent_id in(
(select rent_id from report where report_ispaid = 'N')))
and agent_name = 'meir'
```

	AGENT_ID	AGENT_NAME	AGENT_HIREYEAR	AGENT_SALARY	DEPARTMENT_ID	CITY_ID	SPECIALIZATION
1	6	yaakov	2015	20000.00	1	1	
2	7	meir	2019	18000.00	1	2	
3	8	daniel	2015	25000.00	2	1	
4	9	bat sheva	2018	14000.00	3	8	
5	73237	Kylie	2018	25972.49	21	116	
6	58234	Thelma	2008	11223.16	21	699	
7	67788	Wallace	2007	23639.29	3	286	
8	43839	Terri	2015	13612.42	31	292	
9	36148	Bridget	2020	20203.61	41	628	
10	65285	Russell	2001	35490.73	2	216	
11	67718	Meredith	2017	15380.36	3	1	
12	87483	Humberto	2017	12446.91	3	642	
13	99835	Ray	2021	8651.68	3	262	
14	23484	Julianne	2021	37409.95	6	698	
15	33256	James	2018	7667.37	11	116	
16	31194	Larenz	2014	36793.66	31	731	
17	37792	Charlton	2006	37575.99	41	119	
18	48819	Sissy	2020	39772.20	3	172	
19	99923	Geoff	2020	6940.09	5	388	
20	22231	Chet	2001	28967.68	11	849	

שאילתת עדכון:

השכרה 212 נשמרה בטעות כשולמה, נרצה לעדכן בדוח שההשכרה שמספרה 212 לא שולמה:

השאילתה:

```
update report
set report_ispaid = 'N' where rent_id = 212
```

לפני העדכון:

```
select * from report

update report
set report_ispaid = 'N' where rent_id = 212
```

	REPORT_ID	REPORT_RETURNMODE	REPORT_ISPAID	RENT_ID
1	311	OKAY	Y	212
2	312	OKAY	Y	212
3	313	OKAY	Y	211
4	414	OKAY	Y	314

```
select * from report

update agent
set report_ispaid = 'N' where rent_id = 212
```

	REPORT_ID	REPORT_RETURNMODE	REPORT_ISPAID	RENT_ID
1	311	OKAY	Y	212
2	312	OKAY	Y	212
3	313	OKAY	Y	211
4	414	OKAY	Y	314

לאחר העדכון:

```
select * from report

update report
set report_ispaid = 'N' where rent_id = 212
```

	REPORT_ID	REPORT_RETURNMODE	REPORT_ISPAID	RENT_ID
1	311	OKAY	N	212
2	312	OKAY	N	212
3	313	OKAY	Y	211
4	414	OKAY	Y	314

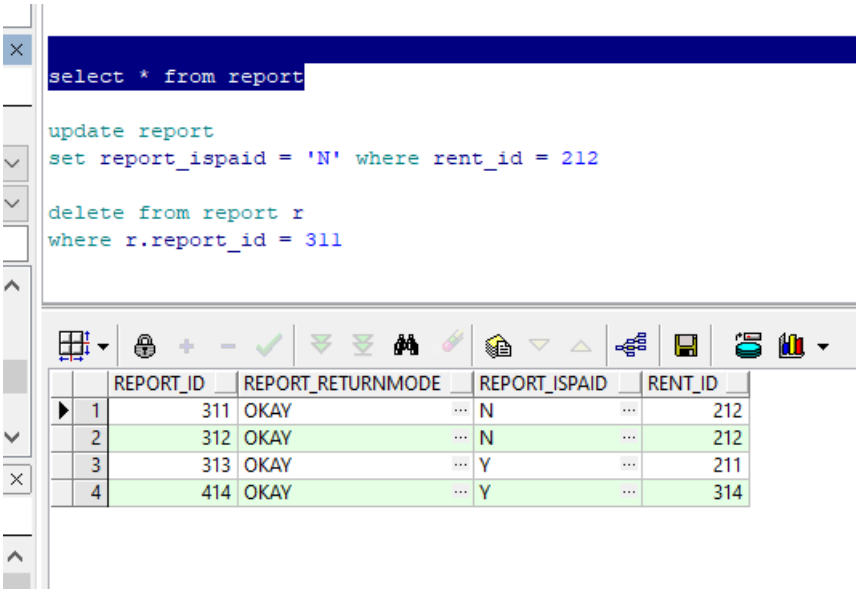
שאילתות מחיקה:

ישנם שני דוחות על השכרה מספר 212, נרצה למחוק אחד מהם:

השאילתה:

```
delete from report r
where r.report_id = 311
```

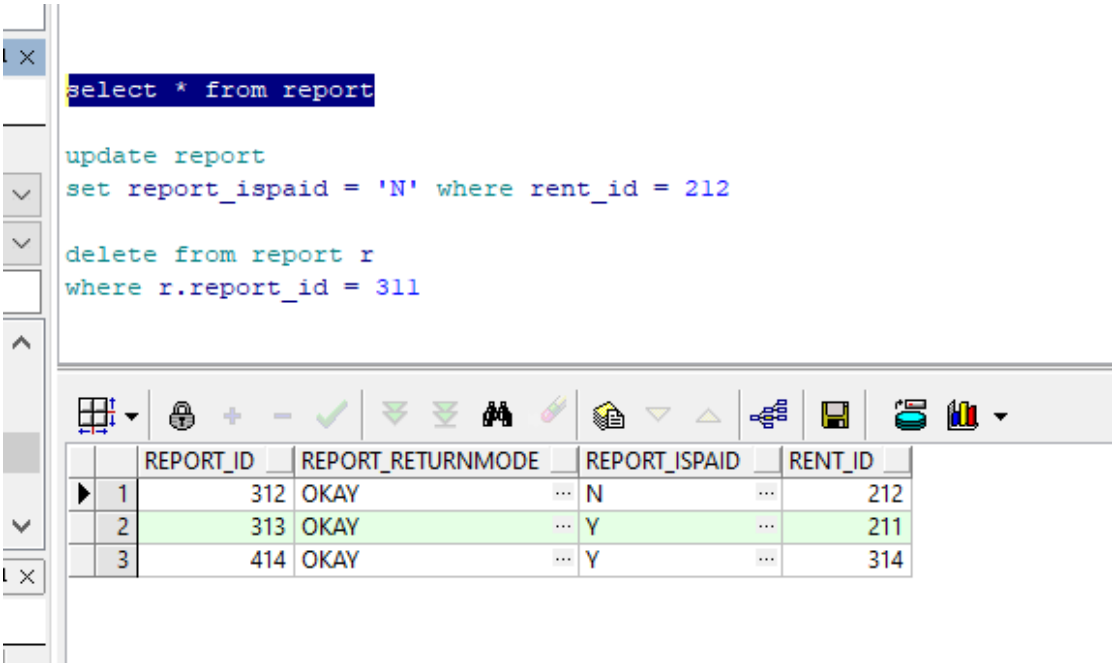
לפני המחיקה:



```
select * from report

update report
set report_ispaid = 'N' where rent_id = 212

delete from report r
where r.report_id = 311
```

	REPORT_ID	REPORT_RETURNMODE	REPORT_ISPAID	RENT_ID
1	311	OKAY	N	212
2	312	OKAY	N	212
3	313	OKAY	Y	211
4	414	OKAY	Y	314

לאחר המחיקה:



```
select * from report

update report
set report_ispaid = 'N' where rent_id = 212

delete from report r
where r.report_id = 311
```

	REPORT_ID	REPORT_RETURNMODE	REPORT_ISPAID	RENT_ID
1	312	OKAY	N	212
2	313	OKAY	Y	211
3	414	OKAY	Y	314

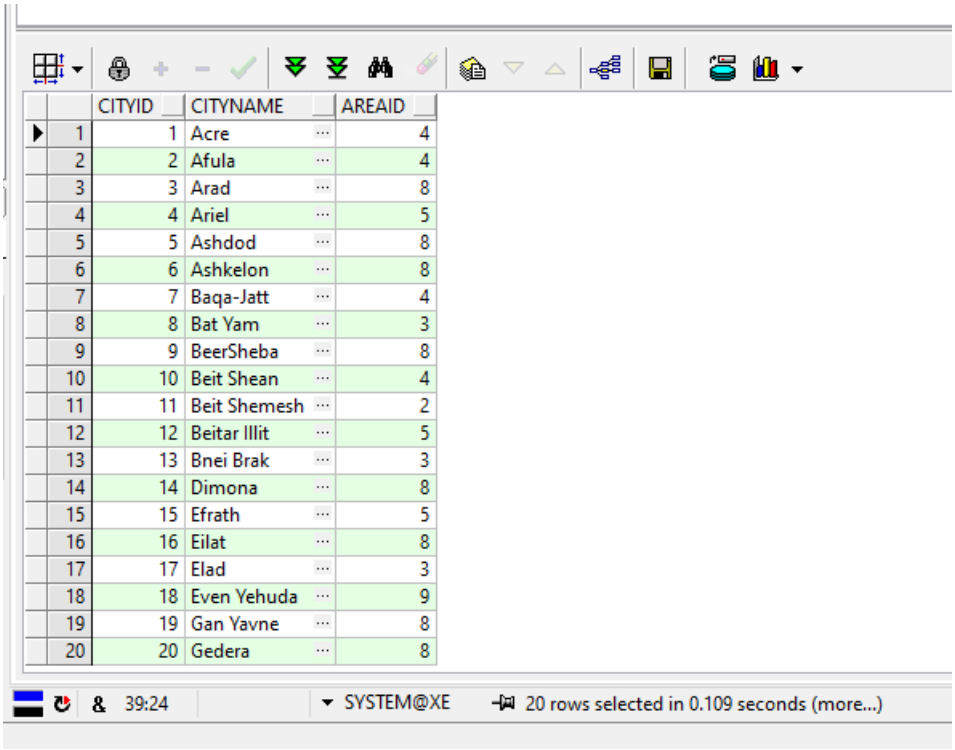
שאילתת מחיקה:

נרצה למחוק את העיר ערד מרשימת הערים כיוון שהארגון אינו עובד בעיר זו יותר:

השאילתה:

```
delete from city c
where c.cityname= 'Arad'
```

לפני המחיקה:

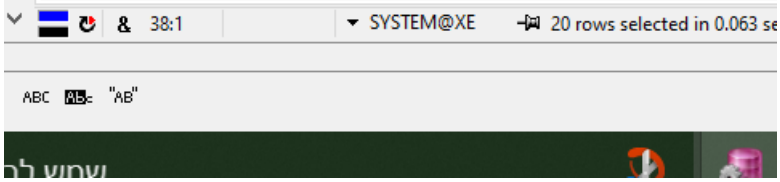


	CITYID	CITYNAME	AREAID
1	1	Acre	4
2	2	Afula	4
3	3	Arad	8
4	4	Ariel	5
5	5	Ashdod	8
6	6	Ashkelon	8
7	7	Baqa-Jatt	4
8	8	Bat Yam	3
9	9	BeerSheba	8
10	10	Beit Shean	4
11	11	Beit Shemesh	2
12	12	Beitar Illit	5
13	13	Bnei Brak	3
14	14	Dimona	8
15	15	Efrath	5
16	16	Eilat	8
17	17	Elad	3
18	18	Even Yehuda	9
19	19	Gan Yavne	8
20	20	Gedera	8

לאחר המחיקה:

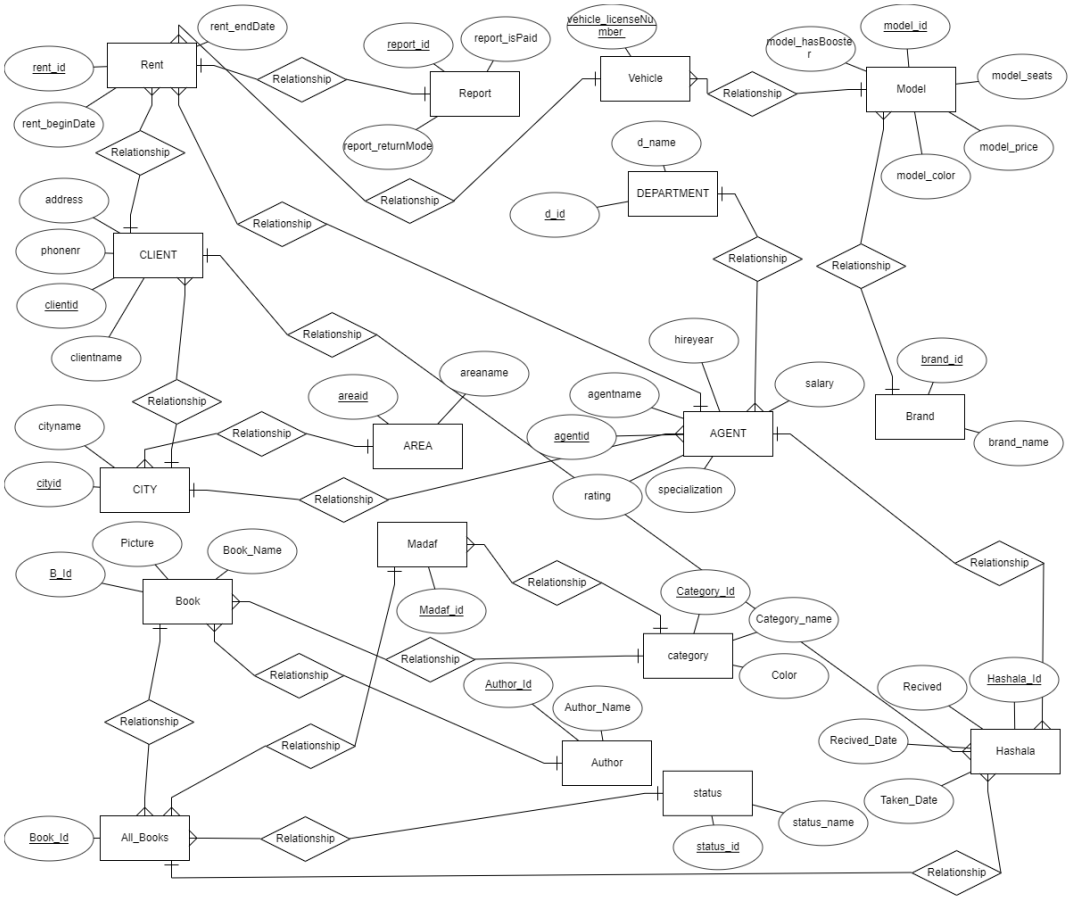
```
select * from city

update report
set report_ispaid = 'N' where rent_id = 212

delete from city c
where c.cityname= 'Arad'
```

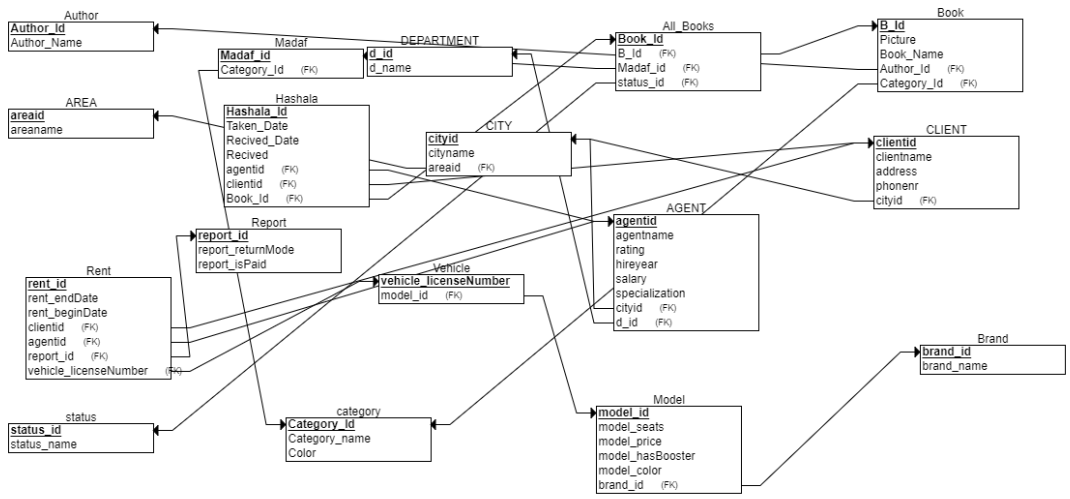
	CITYID	CITYNAME	AREAID
1	1	Acre	4
2	2	Afula	4
3	4	Ariel	5
4	5	Ashdod	8
5	6	Ashkelon	8
6	7	Baqa-Jatt	4
7	8	Bat Yam	3
8	9	BeerSheba	8
9	10	Beit Shean	4
10	11	Beit Shemesh	2
11	12	Beitar Illit	5
12	13	Bnei Brak	3
13	14	Dimona	8
14	15	Efrath	5
15	16	Eilat	8
16	17	Elad	3
17	18	Even Yehuda	9
18	19	Gan Yavne	8
19	20	Gedera	8
20	21	Givat Shmuel	3



שלב 2:
מטלה 5:

חיבור בין שני האגפים השכרת רכב וספרייה
הישויות עיר, סוכן, מחלקה, אזור ולקוח אוחדו משני האגפים כל אחת לישות אחת.
לישות סוכן הוספנו תכונה התמחות, על מנת להבדיל בין סוכן השכרת רכב לבין עובד ספרייה.


rent_endDate
rent_id
Rent
Relationship
report_id
report_isPaid
Report
vehicle_licenseNumber
Vehicle
model_hasBooster
model_id
Model
model_seats
rent_beginDate
Relationship
report_returnMode
Relationship
d_name
DEPARTMENT
d_id
model_price
model_color
address
Relationship
phonenr
CLIENT
clientid
clientname
Relationship
Relationship
Relationship
hireyear
brand_id
areaname
areaid
agentname
salary
Relationship
Brand
cityname
Relationship
AREA
agentid
AGENT
brand_name
cityid
CITY
Relationship
rating
specialization
Picture
Book_Name
Madaf
Relationship
Relationship
B_Id
Book
Madaf_id
Category_Id
Category_name
Relationship
category
Color
Author_Id
Recived
Hashala_Id
Relationship
Author_Name
Relationship
Recived_Date
Hashala
Author
status
Taken_Date
Book_Id
All_Books
Relationship
status_name
status_id
Relationship


Author
Author_Id
Author_Name
Madaf
Madaf_id
Category_Id (FK)
DEPARTMENT
d_id
d_name
All_Books
Book_Id
B_Id (FK)
Madaf_id (FK)
status_id (FK)
Book
B_Id
Picture
Book_Name
Author_Id (FK)
Category_Id (FK)
AREA
areaid
areaname
Hashala
Hashala_Id
Taken_Date
Recived_Date
Recived
agentid (FK)
clientid (FK)
Book_Id (FK)
CITY
cityid
cityname
areaid (FK)
CLIENT
clientid
clientname
address
phonenr
cityid (FK)
AGENT
agentid
agentname
rating
hireyear
salary
specialization
cityid (FK)
d_id (FK)
Report
report_id
report_returnMode
report_isPaid
Rent
rent_id
rent_endDate
rent_beginDate
clientid (FK)
agentid (FK)
report_id (FK)
vehicle_licenseNumber (FK)
Vehicle
vehicle_licenseNumber
model_id (FK)
Brand
brand_id
brand_name
status
status_id
status_name
category
Category_Id
Category_name
Color
Model
model_id
model_seats
model_price
model_hasBooster
model_color
brand_id (FK)

```
CREATE TABLE Author
(
 Author_Name INT NOT NULL,
 Author_Id INT NOT NULL,
 PRIMARY KEY (Author_Id)
);

CREATE TABLE DEPARTMENT
(
 d_name INT NOT NULL,
 d_id INT NOT NULL,
 PRIMARY KEY (d_id)
);

CREATE TABLE AREA
(
 areaname INT NOT NULL,
 areaid INT NOT NULL,
 PRIMARY KEY (areaid)
);

CREATE TABLE CITY
(
 cityid INT NOT NULL,
 cityname INT NOT NULL,
 areaid INT NOT NULL,
 PRIMARY KEY (cityid),
 FOREIGN KEY (areaid) REFERENCES AREA(areaid)
);

CREATE TABLE AGENT
(
 agentname INT NOT NULL,
 rating INT NOT NULL,
 agentid INT NOT NULL,
 hireyear INT NOT NULL,
 salary INT NOT NULL,
 specialization INT NOT NULL,
 cityid INT NOT NULL,
 d_id INT NOT NULL,
```

```
  PRIMARY KEY (agentid),
  FOREIGN KEY (cityid) REFERENCES CITY(cityid),
  FOREIGN KEY (d_id) REFERENCES DEPARTMENT(d_id)
);

CREATE TABLE CLIENT
(
  clientname INT NOT NULL,
  address INT NOT NULL,
  phonenr INT NOT NULL,
  clientid INT NOT NULL,
  cityid INT NOT NULL,
  PRIMARY KEY (clientid),
  FOREIGN KEY (cityid) REFERENCES CITY(cityid)
);

CREATE TABLE Report
(
  report_id INT NOT NULL,
  report_returnMode INT NOT NULL,
  report_isPaid INT NOT NULL,
  PRIMARY KEY (report_id)
);

CREATE TABLE Brand
(
  brand_name INT NOT NULL,
  brand_id INT NOT NULL,
  PRIMARY KEY (brand_id)
);

CREATE TABLE status
(
  status_name INT NOT NULL,
  status_id INT NOT NULL,
  PRIMARY KEY (status_id)
);

CREATE TABLE category
(
  Category_Id INT NOT NULL,
  Category_name INT NOT NULL,
  Color INT NOT NULL,
  PRIMARY KEY (Category_Id)
);

CREATE TABLE Madaf
(
  Madaf_id INT NOT NULL,
  Category_Id INT NOT NULL,
  PRIMARY KEY (Madaf_id),
  FOREIGN KEY (Category_Id) REFERENCES category(Category_Id)
);

CREATE TABLE Book
```

```
(
  Picture INT NOT NULL,
  Book_Name INT NOT NULL,
  B_Id INT NOT NULL,
  Author_Id INT NOT NULL,
  Category_Id INT NOT NULL,
  PRIMARY KEY (B_Id),
  FOREIGN KEY (Author_Id) REFERENCES Author(Author_Id),
  FOREIGN KEY (Category_Id) REFERENCES category(Category_Id)
);

CREATE TABLE Model
(
  model_seats INT NOT NULL,
  model_price INT NOT NULL,
  model_id INT NOT NULL,
  model_hasBooster INT NOT NULL,
  model_color INT NOT NULL,
  brand_id INT NOT NULL,
  PRIMARY KEY (model_id),
  FOREIGN KEY (brand_id) REFERENCES Brand(brand_id)
);

CREATE TABLE All_Books
(
  Book_Id INT NOT NULL,
  B_Id INT NOT NULL,
  Madaf_id INT NOT NULL,
  status_id INT NOT NULL,
  PRIMARY KEY (Book_Id),
  FOREIGN KEY (B_Id) REFERENCES Book(B_Id),
  FOREIGN KEY (Madaf_id) REFERENCES Madaf(Madaf_id),
  FOREIGN KEY (status_id) REFERENCES status(status_id)
);

CREATE TABLE Hashala
(
  Hashala_Id INT NOT NULL,
  Taken_Date INT NOT NULL,
  Recived_Date INT NOT NULL,
  Recived INT NOT NULL,
  agentid INT NOT NULL,
  clientid INT NOT NULL,
  Book_Id INT NOT NULL,
  PRIMARY KEY (Hashala_Id),
  FOREIGN KEY (agentid) REFERENCES AGENT(agentid),
  FOREIGN KEY (clientid) REFERENCES CLIENT(clientid),
  FOREIGN KEY (Book_Id) REFERENCES All_Books(Book_Id)
);

CREATE TABLE Vehicle
(
  vehicle_licenseNumber INT NOT NULL,
  model_id INT NOT NULL,
  PRIMARY KEY (vehicle_licenseNumber),
```

```
 FOREIGN KEY (model_id) REFERENCES Model(model_id)
);

CREATE TABLE Rent
(
 rent_endDate INT NOT NULL,
 rent_beginDate INT NOT NULL,
 rent_id INT NOT NULL,
 clientid INT NOT NULL,
 agentid INT NOT NULL,
 report_id INT NOT NULL,
 vehicle_licenseNumber INT NOT NULL,
 PRIMARY KEY (rent_id),
 FOREIGN KEY (clientid) REFERENCES CLIENT(clientid),
 FOREIGN KEY (agentid) REFERENCES AGENT(agentid),
 FOREIGN KEY (report_id) REFERENCES Report(report_id),
 FOREIGN KEY (vehicle_licenseNumber) REFERENCES Vehicle(vehicle_licenseNumber)
                                                                                );
```

מהלך האינטגרציה:

נבחר את כל הרשומות שנמצאות בישות עיר של האגף ספרייה ולא נמצאות באגף השכרת רכב, אם אכן קיימות כאלה – נכניס אותן לישות עיר של האגף השכרת רכב:

```
select * from client a where a.clientid not in(select client_id from
client1);

insert into client1 t (t.client_id, t.client_name, t.client_phonenr,
t.client_address, t.city_id)
select s.clientid, s.clientname, s.phonenr, s.address, s.cityid
from client s
             where s.clientid not in (select client_id from client1);
```

נבחר את כל הרשומות שנמצאות בישות מחלקה של האגף ספרייה ולא נמצאות באגף השכרת רכב, גילינו שאין כאלה ולכן נמשיך הלאה:

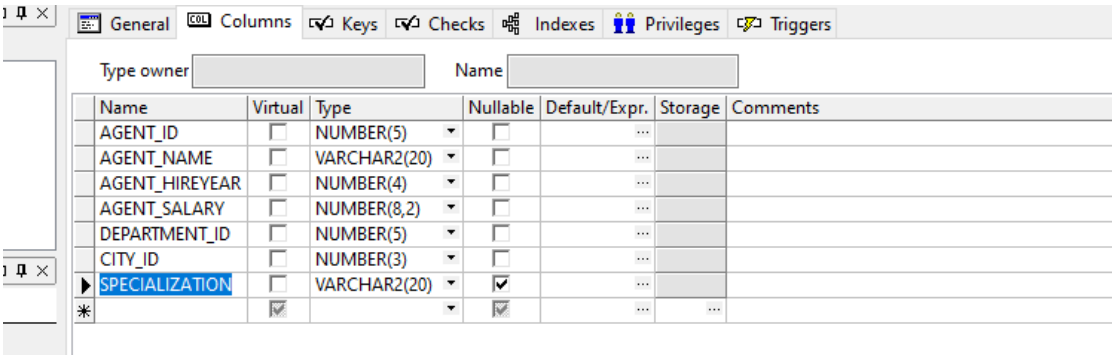
```
select * from department a where a.d_id not in(select department_id
from department1);
```

נבחר את כל הרשומות שנמצאות בישות אזור של האגף ספרייה ולא נמצאות באגף השכרת רכב, אם אכן קיימות כאלה – נכניס אותן לישות עיר של האגף השכרת רכב:

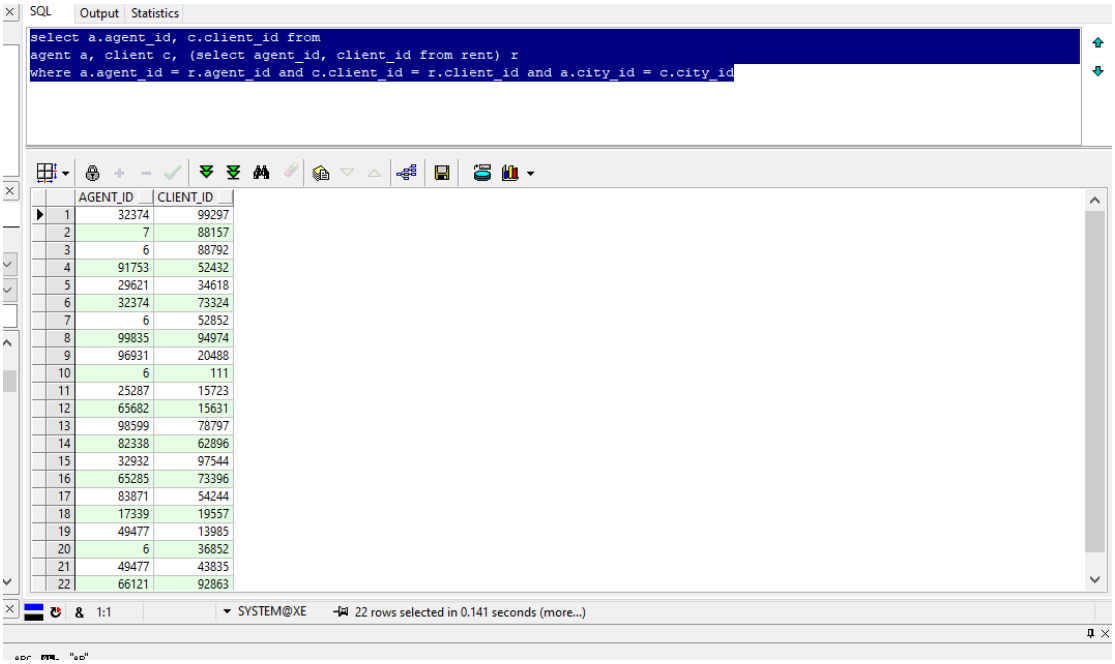
```
select count(*) from area;
select count(*) from area1;

select * from area a where a.areaid not in(select area_id from
area1);

insert into area1 t (t.area_id, t.area_name)
select s.areaid, s.areaname
from area s
where s.areaid not in (select area_id from area1);
```

כך ביצענו עבור כל הישויות החופפות: עיר, אזור, לקוח, מחלקה, סוכן.

נוסיף את התכונה התמחות לישות סוכן:

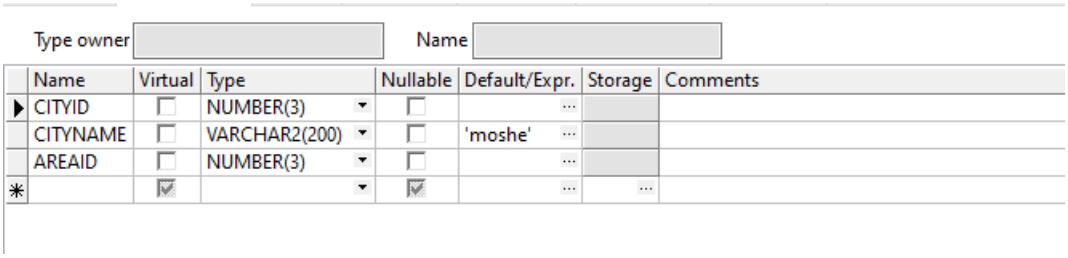


השאילתות אכן פועלות גם לאחר האינטגרציה:



אילוצים:

הוספנו ערך דיפולטיבי עבור עמודת השם:




```
alter table model
add constraint check_model_seats
                                    check ( model_seats > 4 );


alter table model
add constraint chek_C
                                    check (city_name != 'Acre' );
```

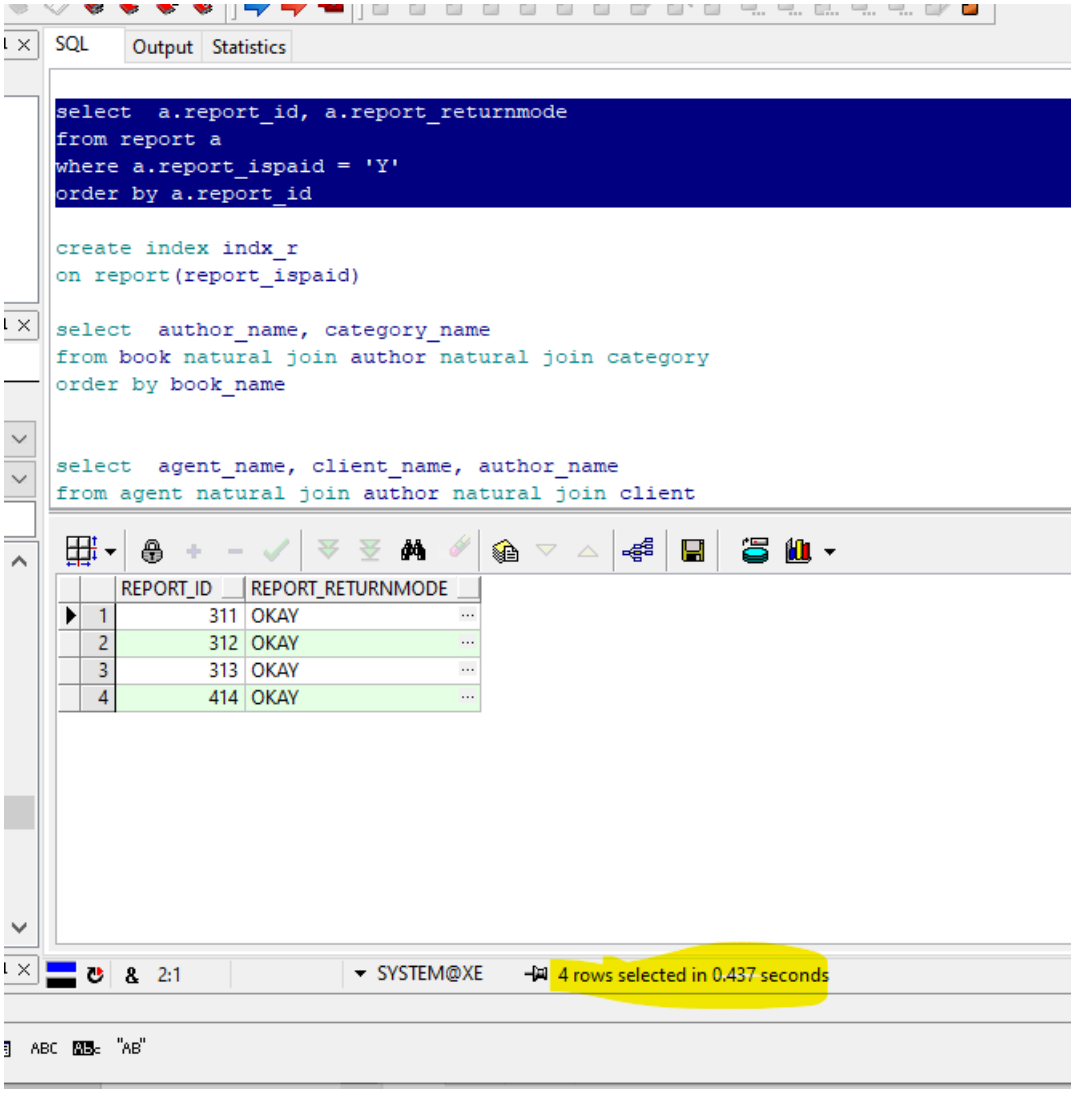
שלב 3:

מטלה 6:

שאילתות:

```
select  a.report_id, a.report_returnmode
from report a
where a.report_ispaid = 'Y'
order by a.report_id
```

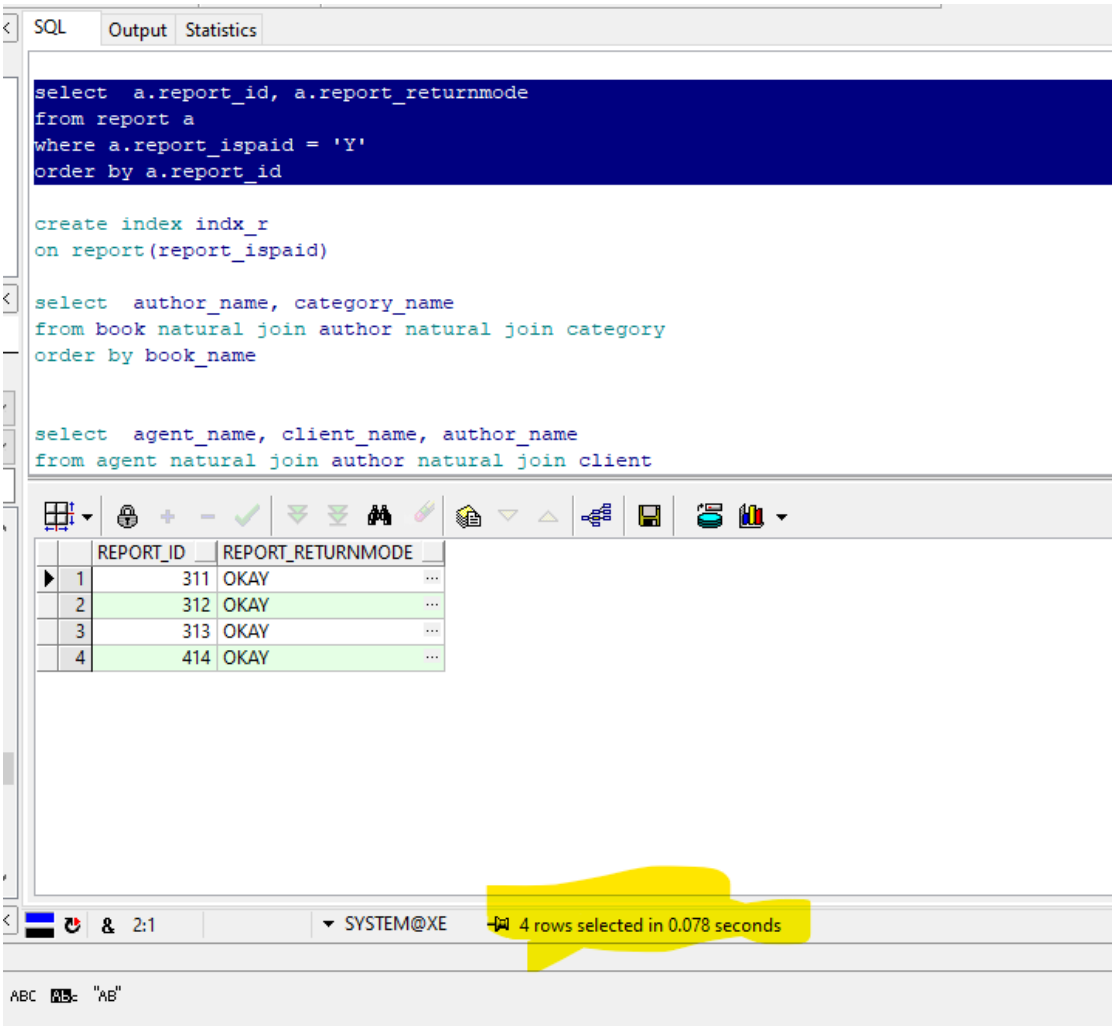
לפני יצירת האינדקס:



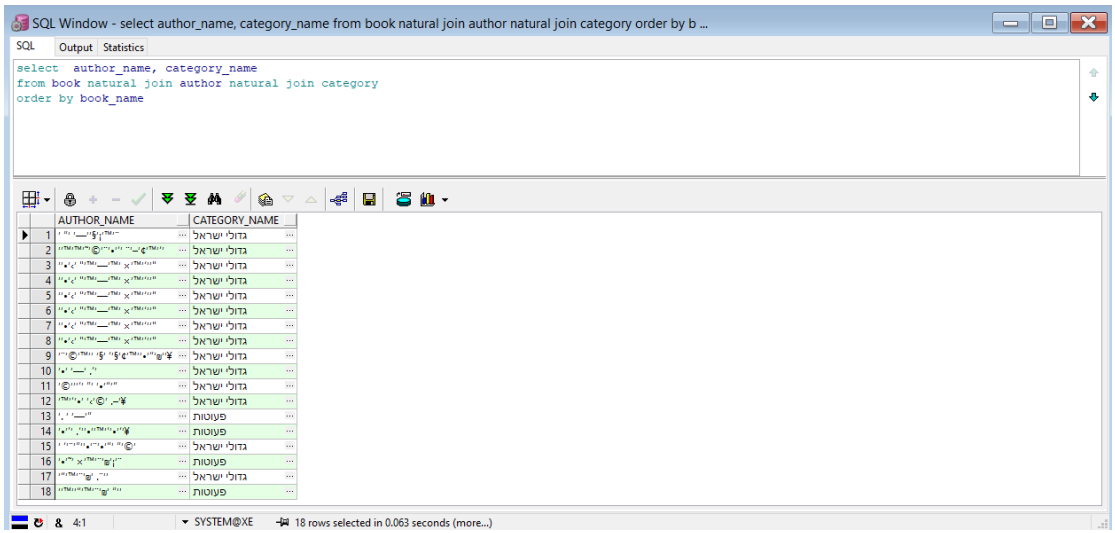
האינדקס:

```
create index indx_r
on report(report_ispaid)
```

לאחר יצירת האינדקס:

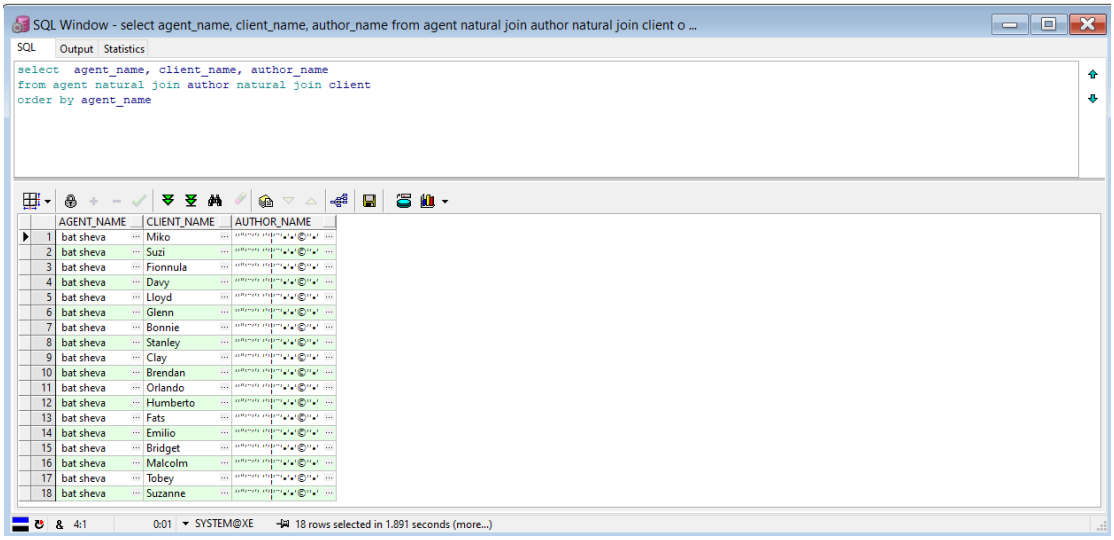



```
select  author_name, category_name
from book natural join author natural join category
order by book_name
```

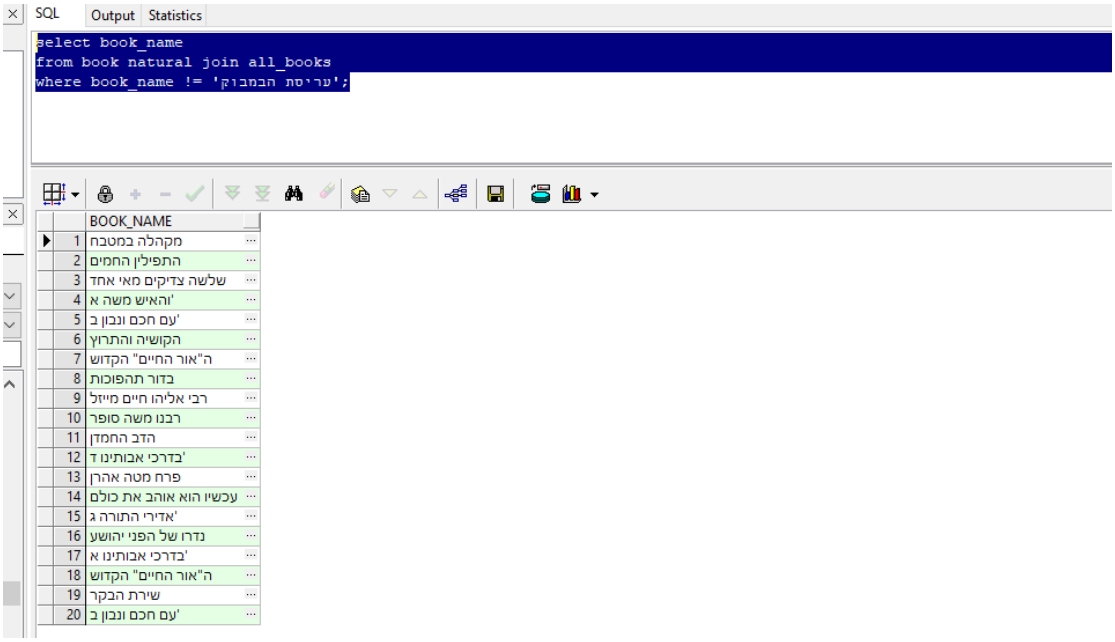



```
select  agent_name, client_name, author_name
```

```
from agent natural join author natural join client
order by agent_name
```




```
select book_name
from book natural join all_books
';עריסת הבמבוקwhere book_name != '
```

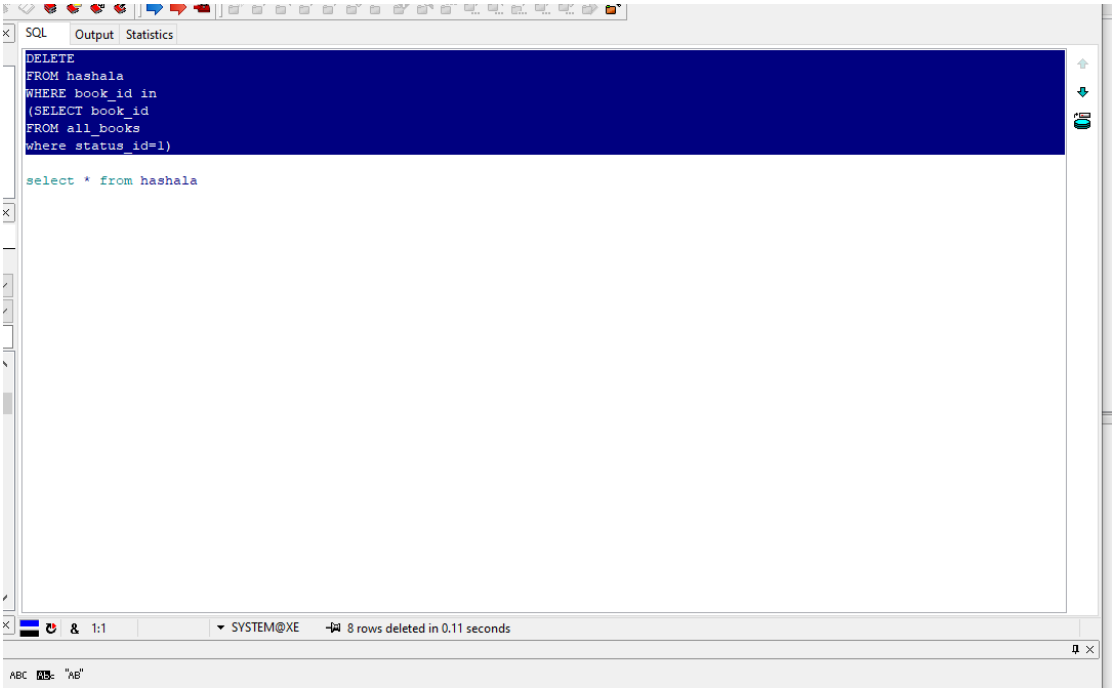


שאילתת מחיקה:

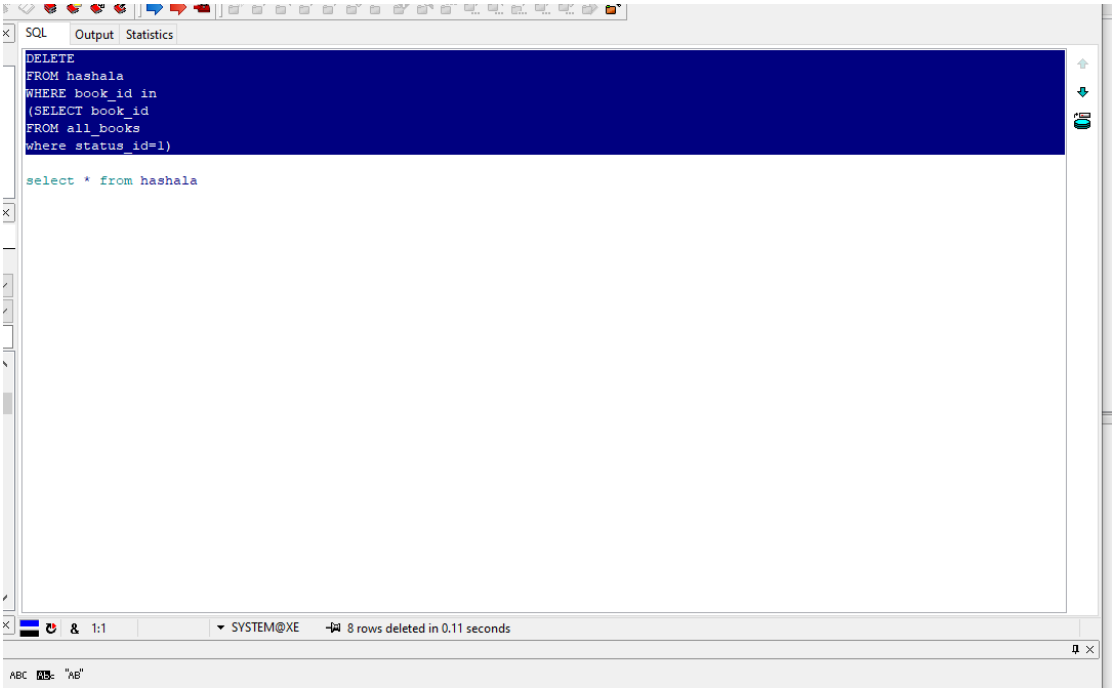
```
DELETE
FROM hashala
WHERE book_id in
(SELECT book_id
FROM all_books
```

```
where status_id=1)
```

לפני המחיקה:

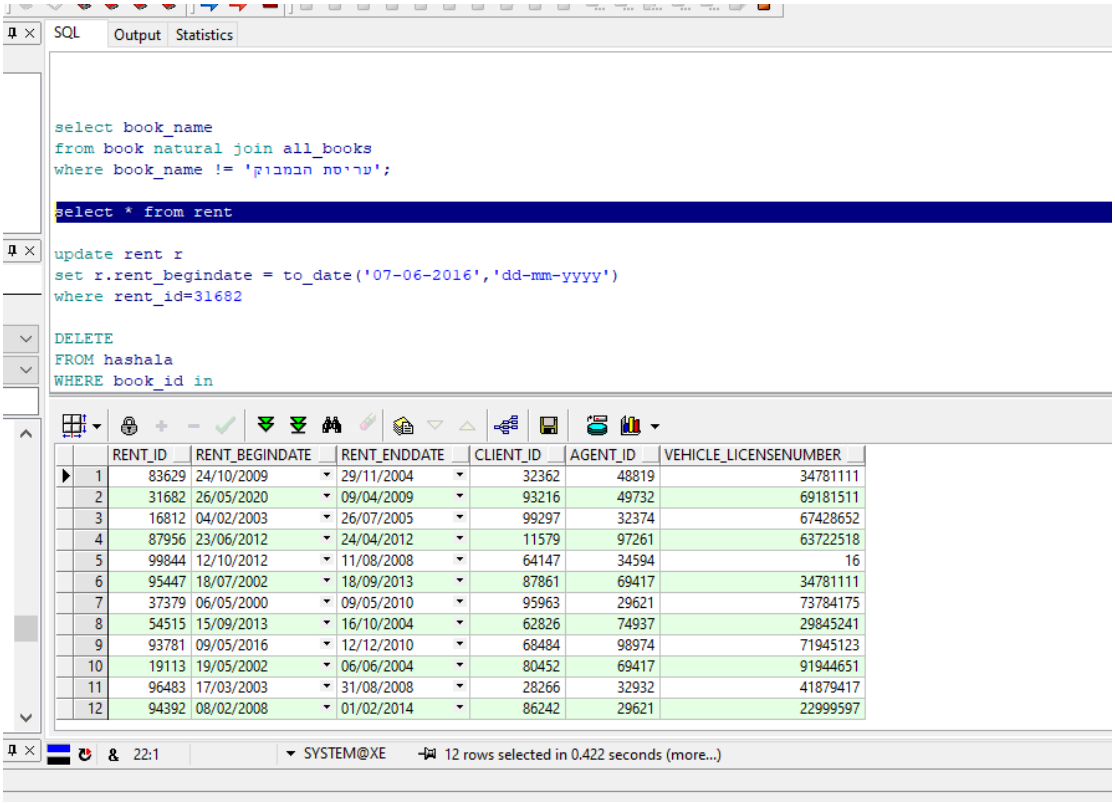


לאחר המחיקה:

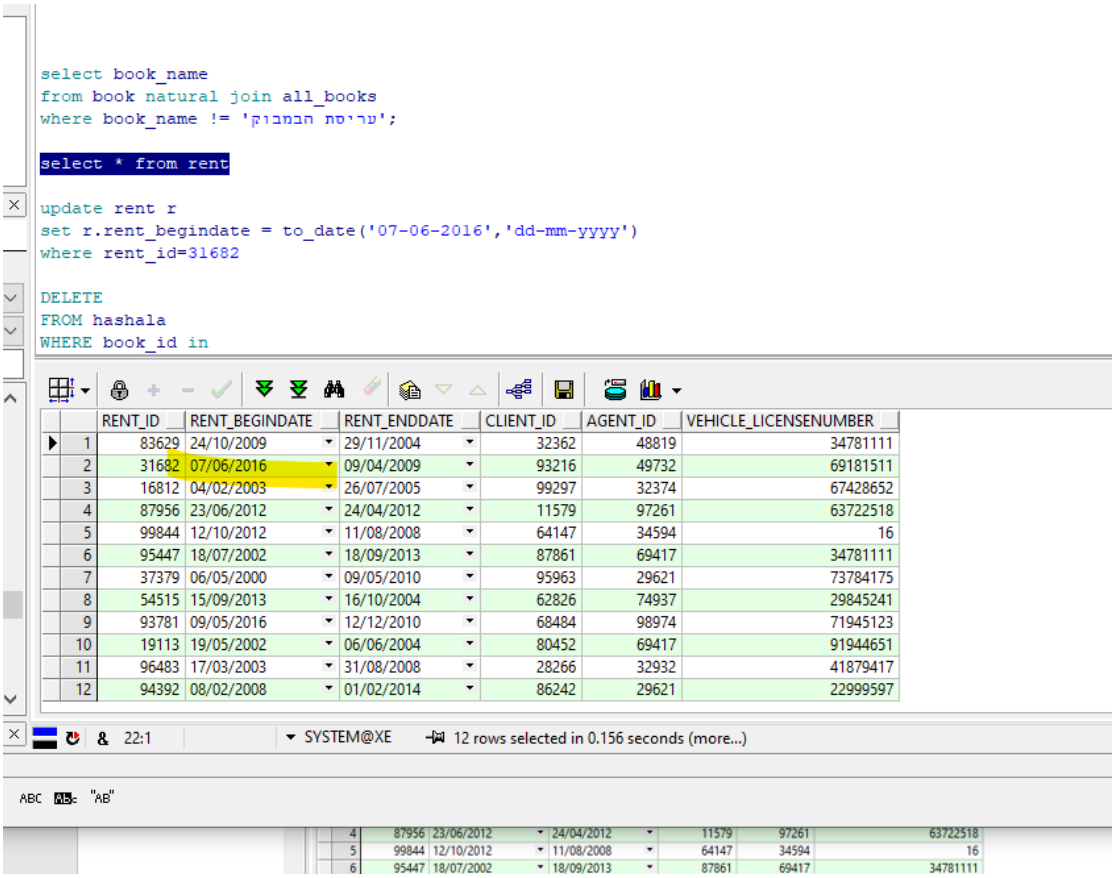


עדכון:

```
update rent r
set r.rent_begindate = to_date('07-06-2016','dd-mm-yyyy')
where rent_id=31682
```



לאחר העדכון:



```
select book_name
from book natural join all_books
where book_name != 'ערוסת התמרוק';

select * from rent

update rent r
set r.rent_begindate = to_date('07-06-2016','dd-mm-yyyy')
where rent_id=31682

DELETE
FROM hashala
WHERE book_id in
```

	RENT_ID	RENT_BEGINDATE	RENT_ENDDATE	CLIENT_ID	AGENT_ID	VEHICLE_LICENSENUMBER
1	83629	24/10/2009	29/11/2004	32362	48819	34781111
2	31682	07/06/2016	09/04/2009	93216	49732	69181511
3	16812	04/02/2003	26/07/2005	99297	32374	67428652
4	87956	23/06/2012	24/04/2012	11579	97261	63722518
5	99844	12/10/2012	11/08/2008	64147	34594	16
6	95447	18/07/2002	18/09/2013	87861	69417	34781111
7	37379	06/05/2000	09/05/2010	95963	29621	73784175
8	54515	15/09/2013	16/10/2004	62826	74937	29845241
9	93781	09/05/2016	12/12/2010	68484	98974	71945123
10	19113	19/05/2002	06/06/2004	80452	69417	91944651
11	96483	17/03/2003	31/08/2008	28266	32932	41879417
12	94392	08/02/2008	01/02/2014	86242	29621	22999597

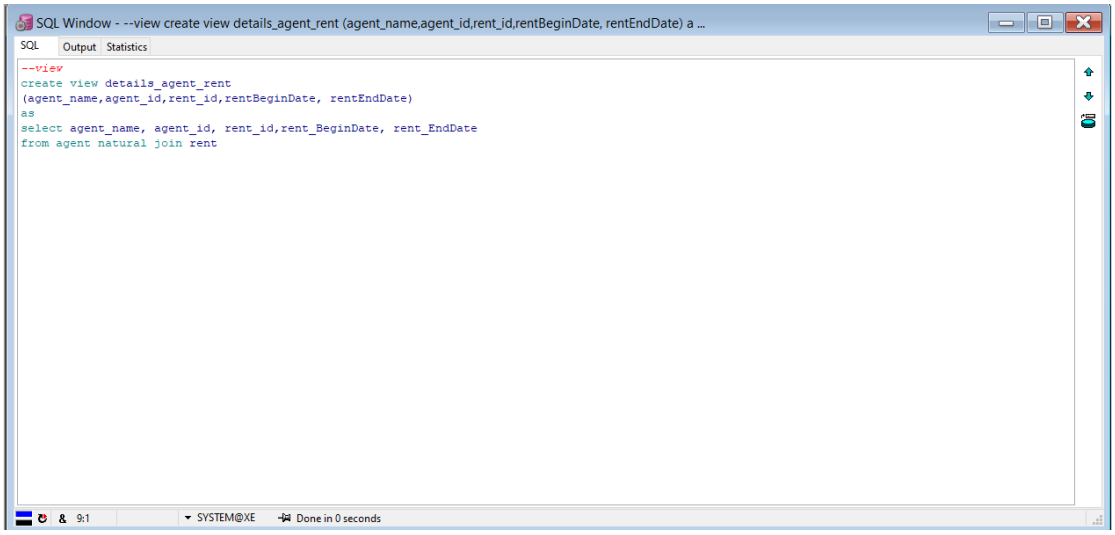
שלב 3:

מטלה 7:

מבטים:

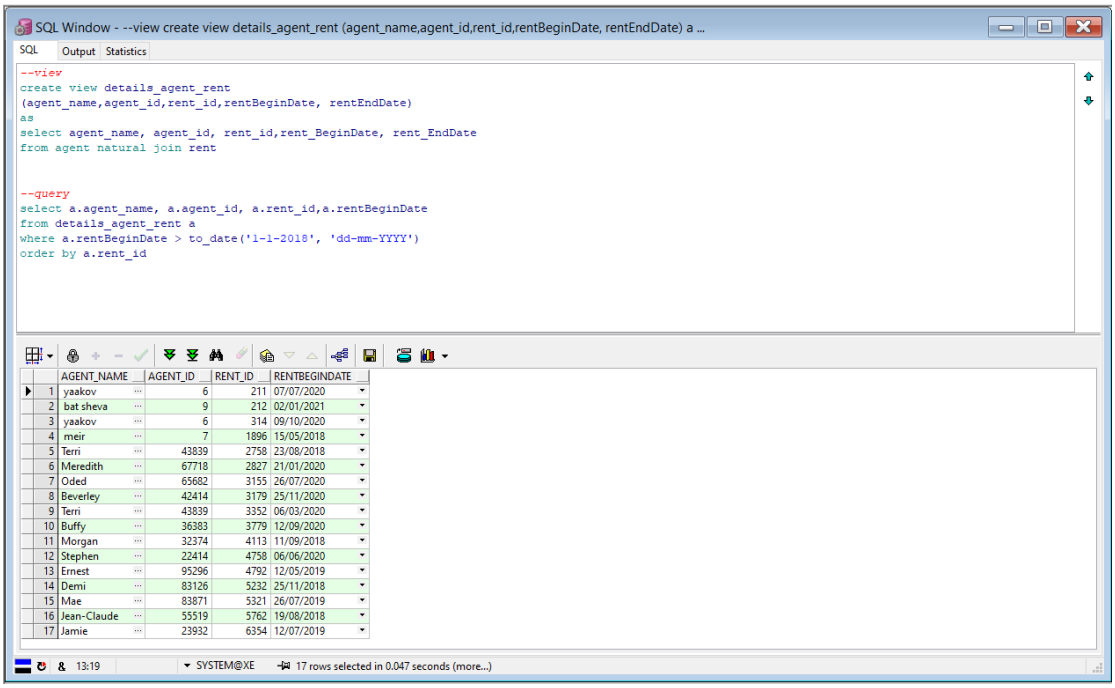
נרצה לקבל מבט על הסוכנים וזמני ההשכרות שביצעו:

```
--view
create view details_agent_rent
(agent_name,agent_id,rent_id,rentBeginDate, rentEndDate)
as
select agent_name, agent_id, rent_id,rent_BeginDate, rent_EndDate
from agent natural join rent
```



שאילתות על המבט:
נבחר מתוך המבט רק את פרטי הסוכנים שביצעו השכרות החל מ2018 והלאה:

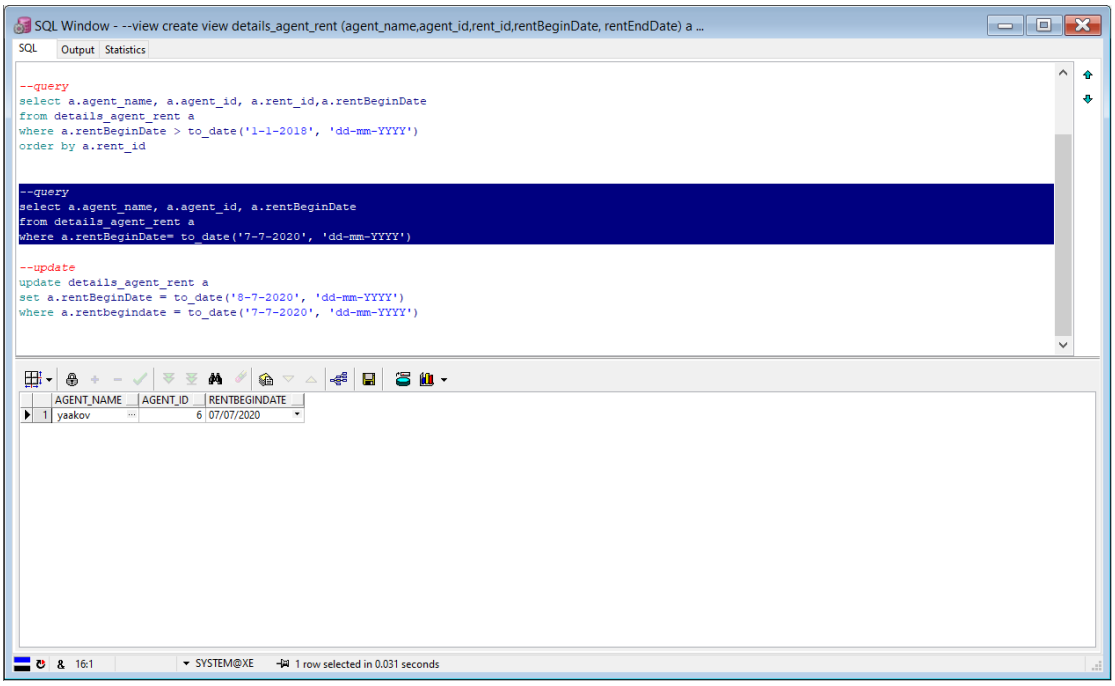
```
--query
select a.agent_name, a.agent_id, a.rent_id,a.rentBeginDate
from details_agent_rent a
where a.rentBeginDate > to_date('1-1-2018', 'dd-mm-YYYY')
order by a.rent_id
```



נבצע עדכון:
לפני העדכון, נבחר את ההשכרות שבוצעו בתאריך המוזכר, אך מדובר בטעות כיוון שבתאריך זה הייתה חופשה לעובדים וכל ההשכרות בוצעו יום למחרת:

```
--query
select a.agent_name, a.agent_id, a.rentBeginDate
```

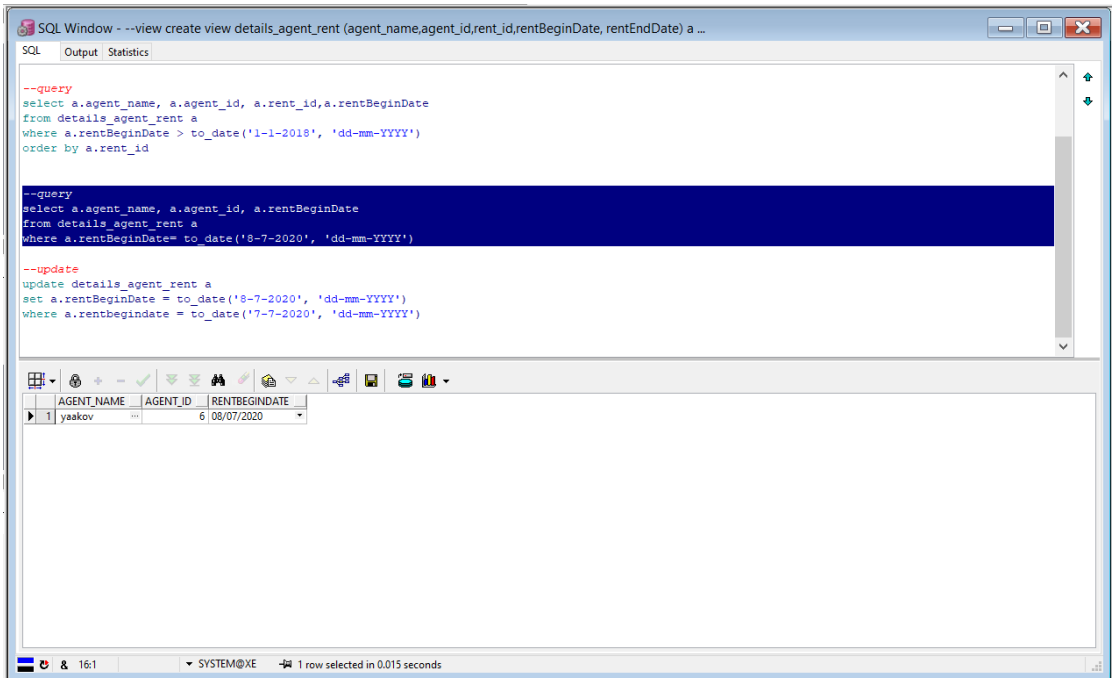
```
from details_agent_rent a
where a.rentBeginDate= to_date('7-7-2020', 'dd-mm-YYYY')
```



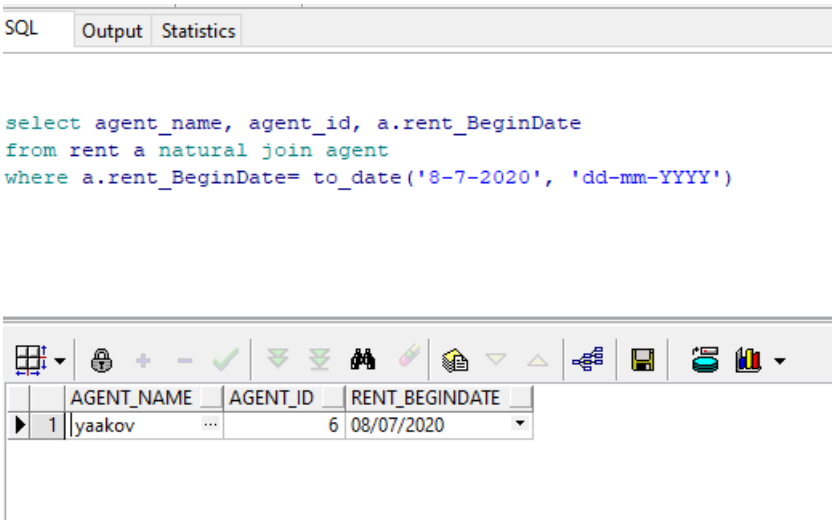
לכן, נעדכן את תאריך ההשכרה ליום למחרת:

```
--update
update details_agent_rent a
set a.rentBeginDate = to_date('8-7-2020', 'dd-mm-YYYY')
where a.rentbegindate = to_date('7-7-2020', 'dd-mm-YYYY')
```

לאחר העדכון:



העדכון התבצע גם על הטבלה, כיוון שהמבט שיצרנו לא מורכב מאילוצים שמונעים ממנו לעדכן את טבלת המקור:

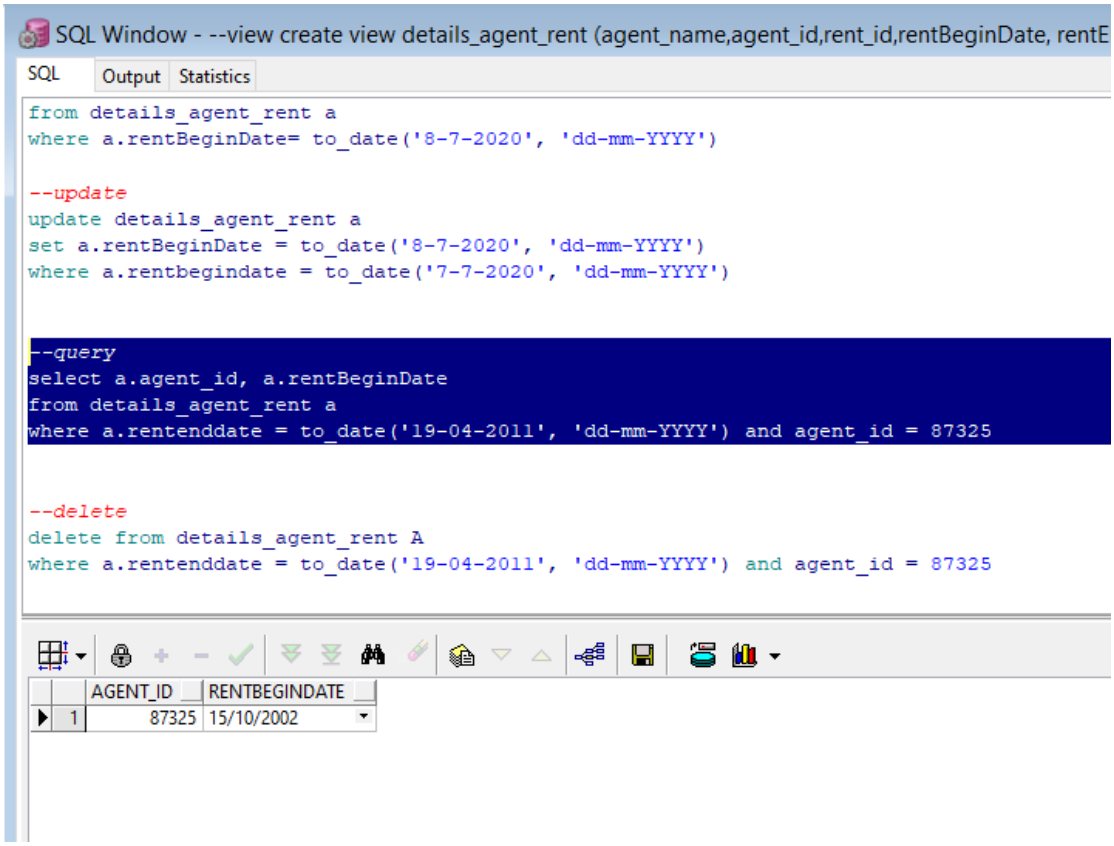


שאילתת מחיקה:

נרצה למחוק מהמבט את ההשכרות שבוצעו ע"י סוכן מסויים והוחזרו בתאריך מסויים, כיוון שסוכן זה פוטר באותו יום, והשכרותיו הועברו לסוכן אחר:

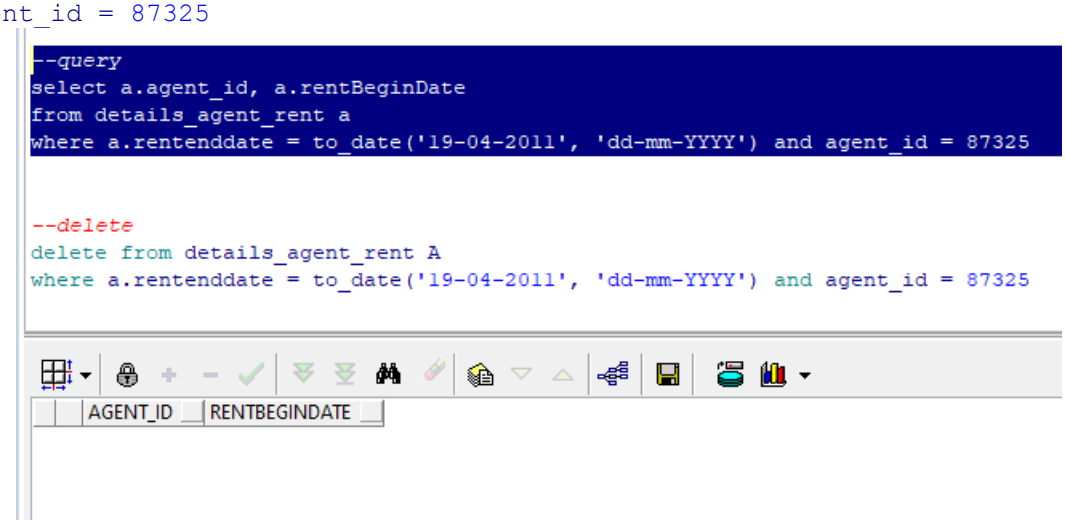
לפני המחיקה:

```
--query
select a.agent_id, a.rentBeginDate
from details_agent_rent a
where a.rentenddate = to_date('19-04-2011', 'dd-mm-YYYY') and
agent_id = 87325
```



לאחר המחיקה:

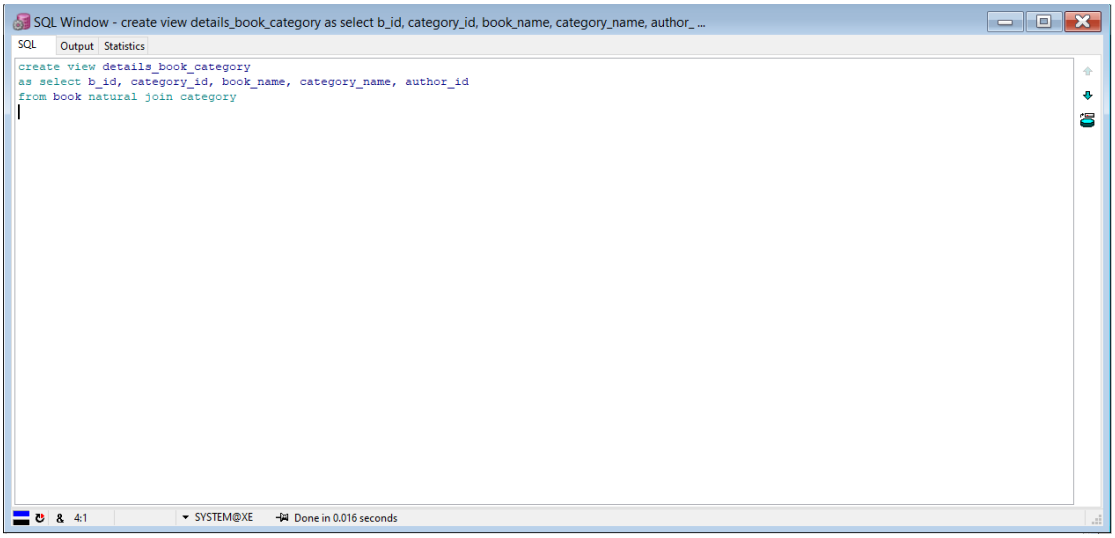
```
--delete
delete from details_agent_rent A
where a.rentenddate = to_date('19-04-2011', 'dd-mm-YYYY') and
agent_id = 87325
```



מבט על הספרים:

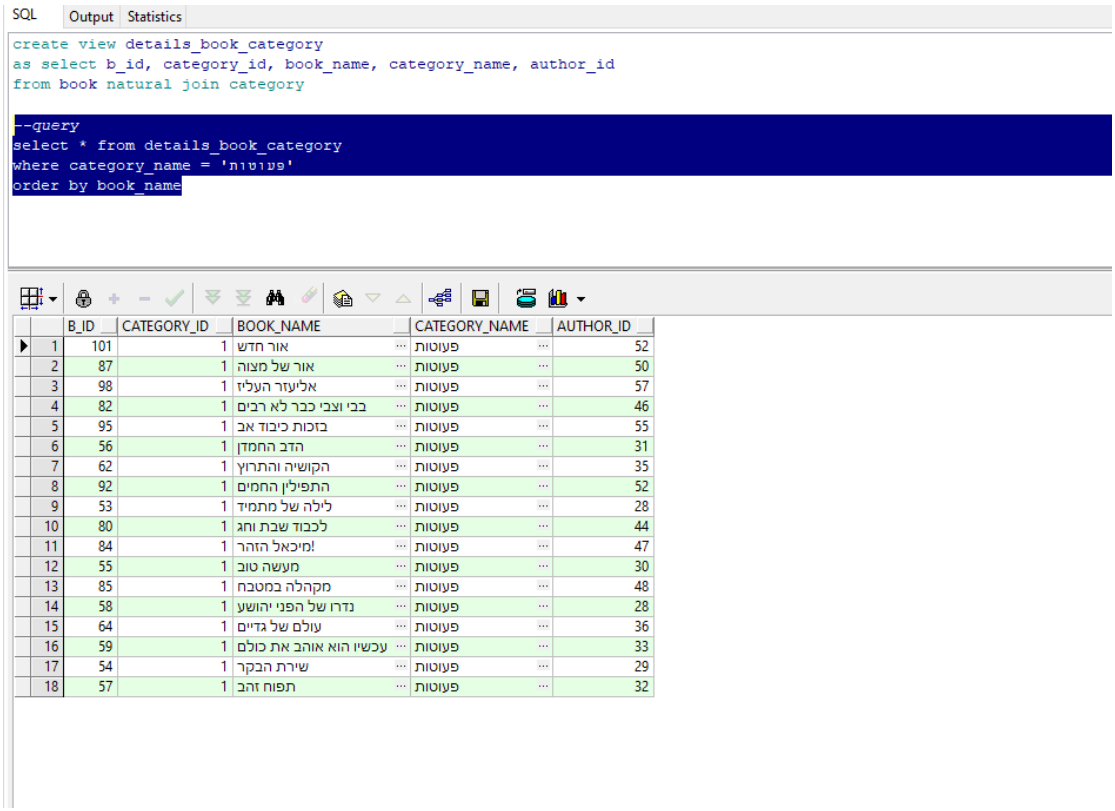
נרצה לקבל מבט על הספרים והקטגוריה שלהם:

```
create view details_book_category
as select b_id, category_id, book_name, category_name, author_id
from book natural join category
```



נרצה לשלוף מהמבט את כל הנתונים אודות הספרים שהם בקטגוריית פעוטות, ולסדר אותם על פי שם הספר:

```
--query
select * from details_book_category
where category_name = 'פעוטות'
order by book_name
```



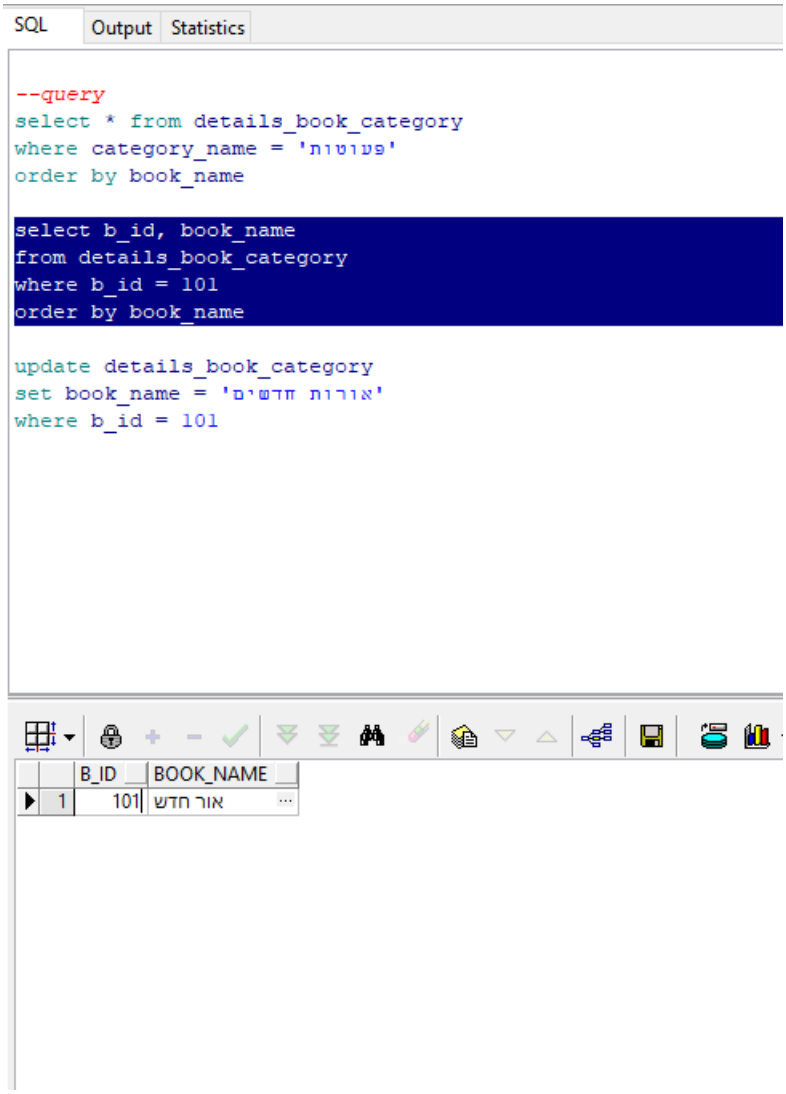
	B_ID	CATEGORY_ID	BOOK_NAME	CATEGORY_NAME	AUTHOR_ID
1	101	1	אור חדש	פעוטות	52
2	87	1	אור של מצוה	פעוטות	50
3	98	1	אליעזר העליז	פעוטות	57
4	82	1	בבי וצבי כבר לא רבים	פעוטות	46
5	95	1	בזכות כיבוד אב	פעוטות	55
6	56	1	הדב החמדן	פעוטות	31
7	62	1	הקושיה והתרוץ	פעוטות	35
8	92	1	התפילין החמים	פעוטות	52
9	53	1	לילה של מתמיד	פעוטות	28
10	80	1	לכבוד שבת וחג	פעוטות	44
11	84	1	מיכאל הזהר!	פעוטות	47
12	55	1	מעשה טוב	פעוטות	30
13	85	1	מקהלה במטבח	פעוטות	48
14	58	1	נדרו של הפני יהושע	פעוטות	28
15	64	1	עולם של גדיים	פעוטות	36
16	59	1	עכשיו הוא אוהב את כולם	פעוטות	33
17	54	1	שירת הבקר	פעוטות	29
18	57	1	תפוח זהב	פעוטות	32

שם הספר שמספר 101 נשמר בטעות כ"אור חדש" במקום "אורות חדשים", נרצה לעדכן זאת:

```
select b_id, book_name
from details_book_category
```

```
where b_id = 101
order by book_name
```

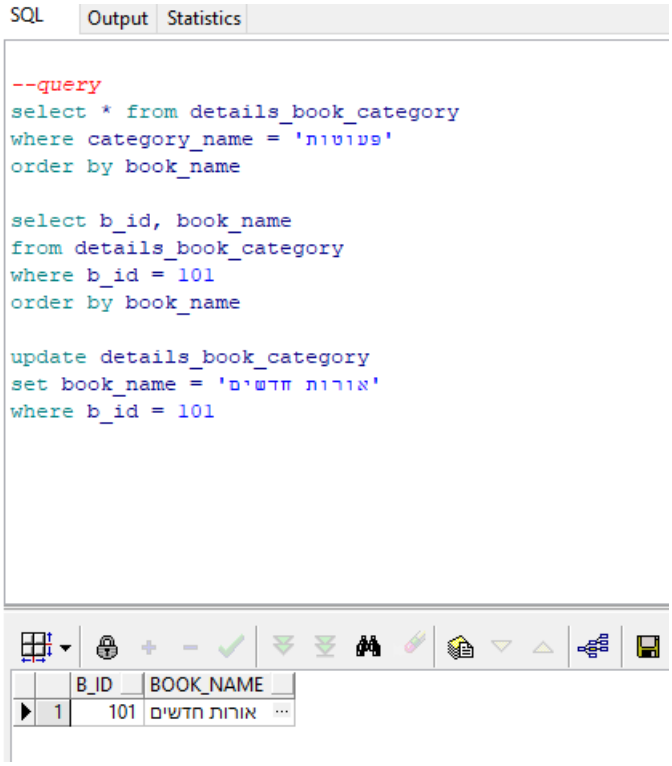
לפני העדכון:



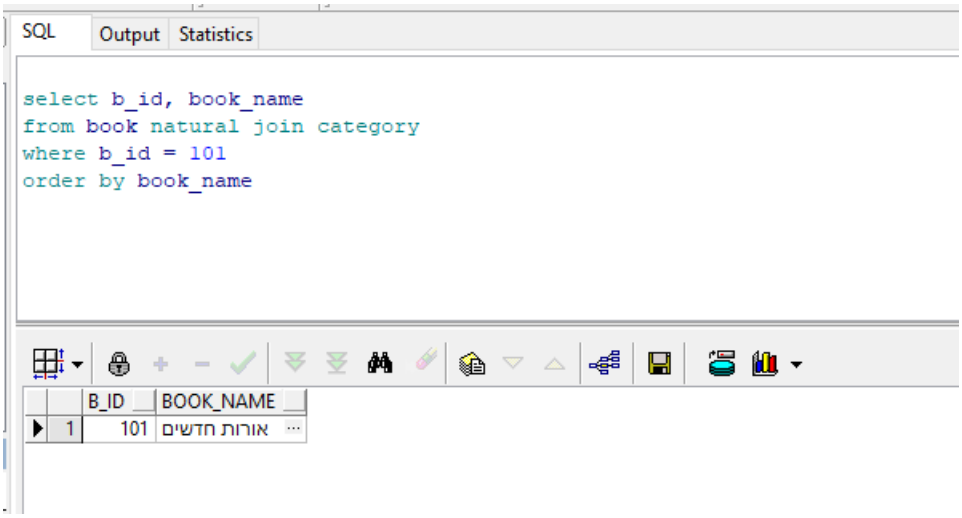
שאילתת העדכון:

```
update details_book_category
set book_name = 'אורות חדשים'
where b_id = 101
```

לאחר העדכון:



העדכון התבצע גם על הטבלה, כיוון שהמבט שיצרנו מורכב רק מאילוצים שמאפשרים לו לעדכן את טבלת המקור:



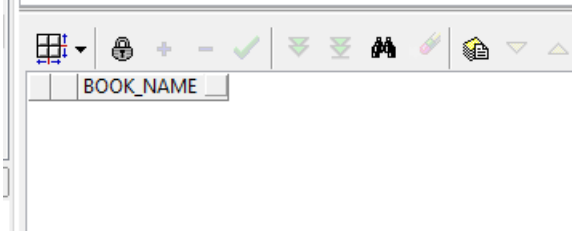
שאילתת מחיקה:

נרצה למחוק את הספר לילה של מתמיד שמספרו 53 כיוון שהוא נאסר להפצה:

```
delete from details_book_category
where b_id = 53
```

לאחר המחיקה:

```
select book_name
from details_book_category
where b_id = 53

delete from details_book_category
where b_id = 53
```

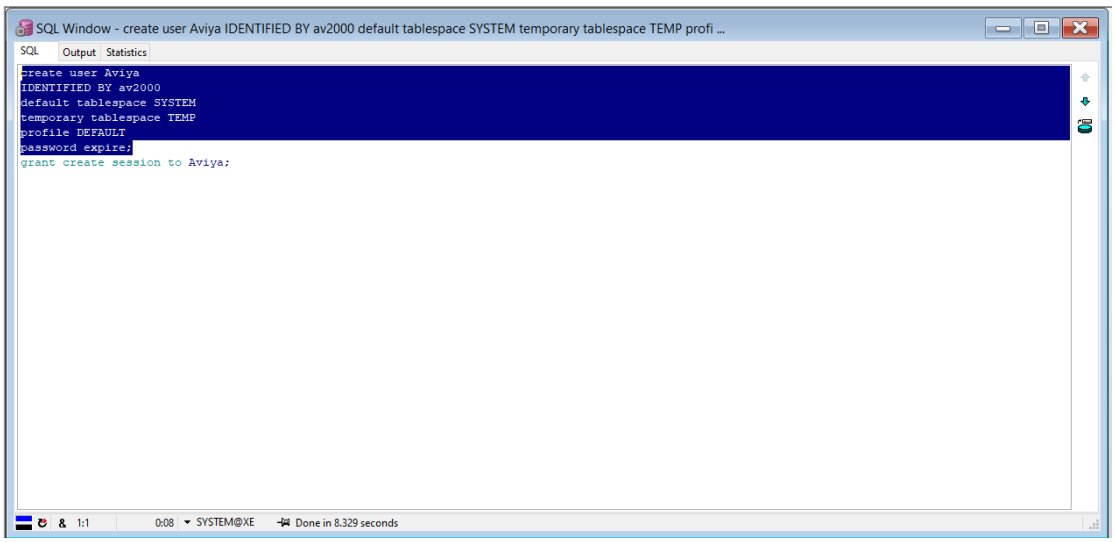


נבצע grant:

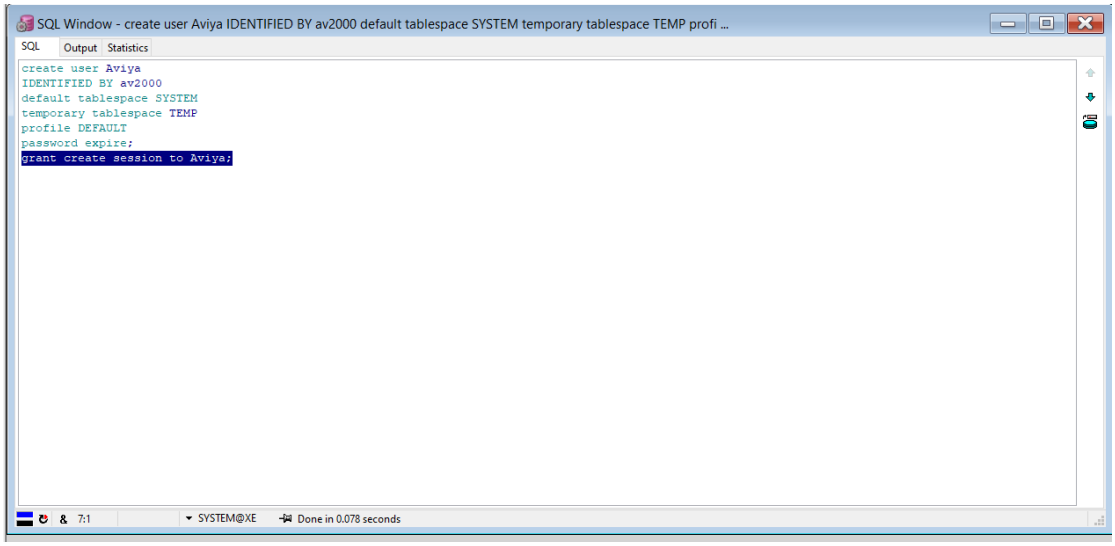
נתינת הרשאה:

ניצור את המשתמש אביה:

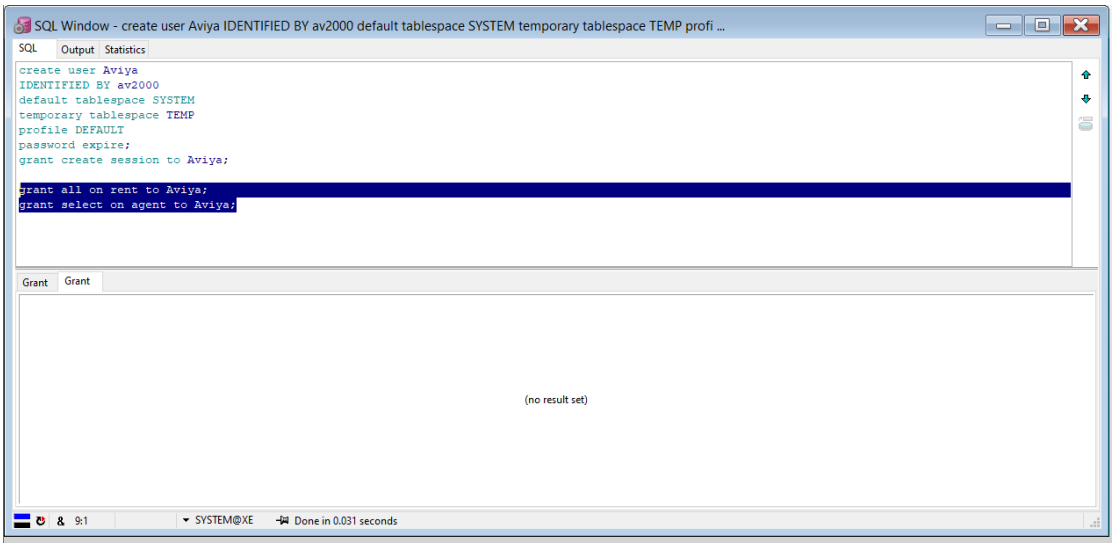
```
create user Aviya
IDENTIFIED BY av2000
default tablespace SYSTEM
temporary tablespace TEMP
profile DEFAULT
password expire;
grant create session to Aviya;

grant all on rent to Aviya;
grant select on agent to Aviya;
```



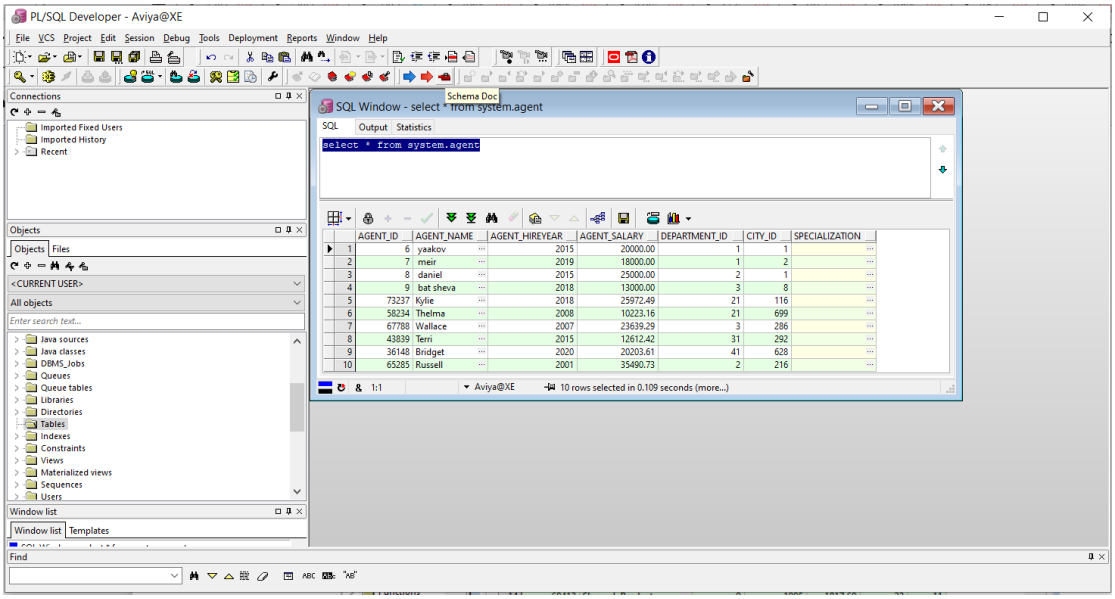
ניתן לו הרשאת גישה:



ניתן לו הרשאת גישה לטבלאות שנבחר:

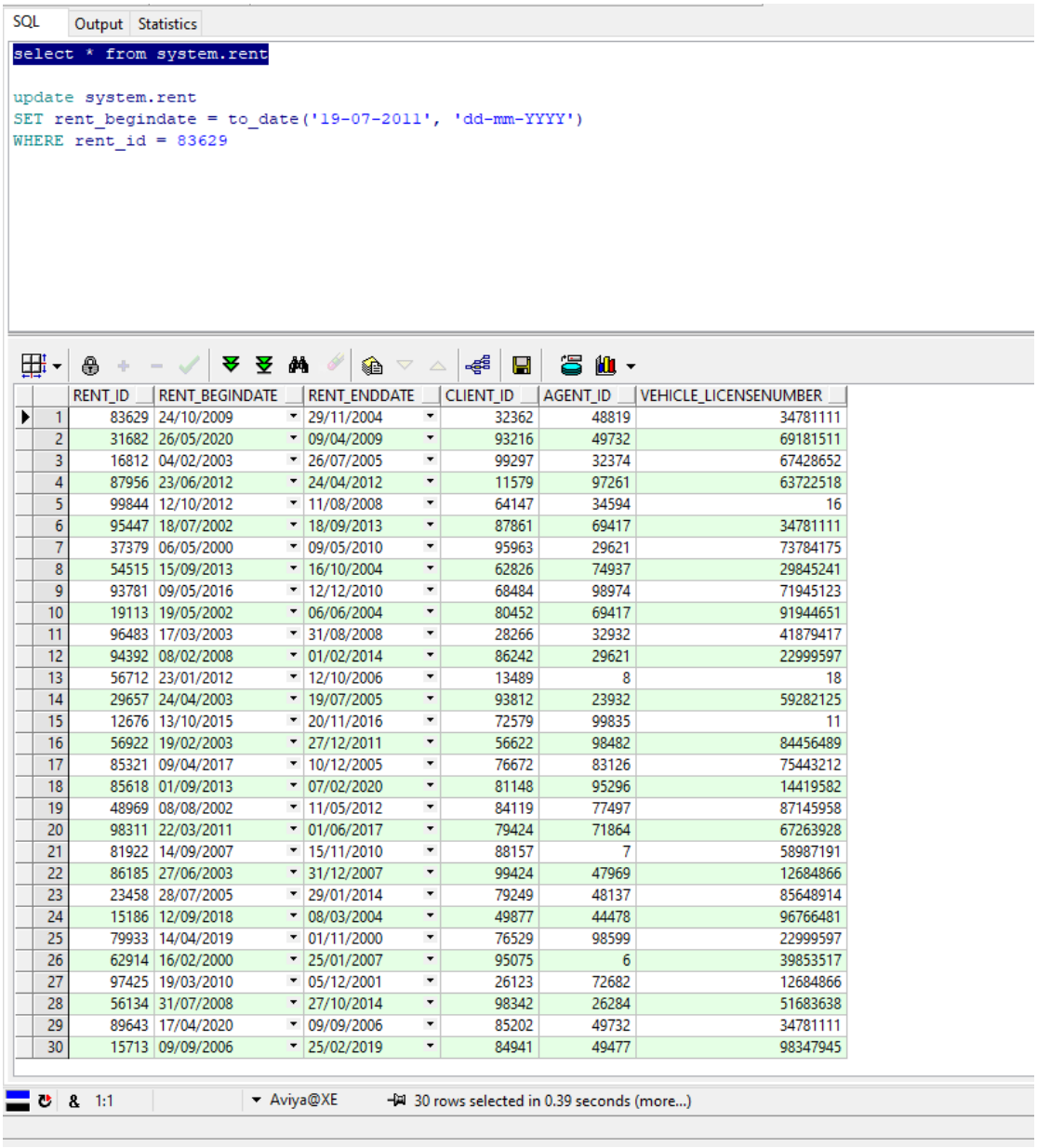


נראה שאכן אביה קיבלה את ההרשאה ויש לה גישה לטבלה:



נבצע עדכון לטבלה השכרה ע"י המשתמש אביה:

לפני העדכון:

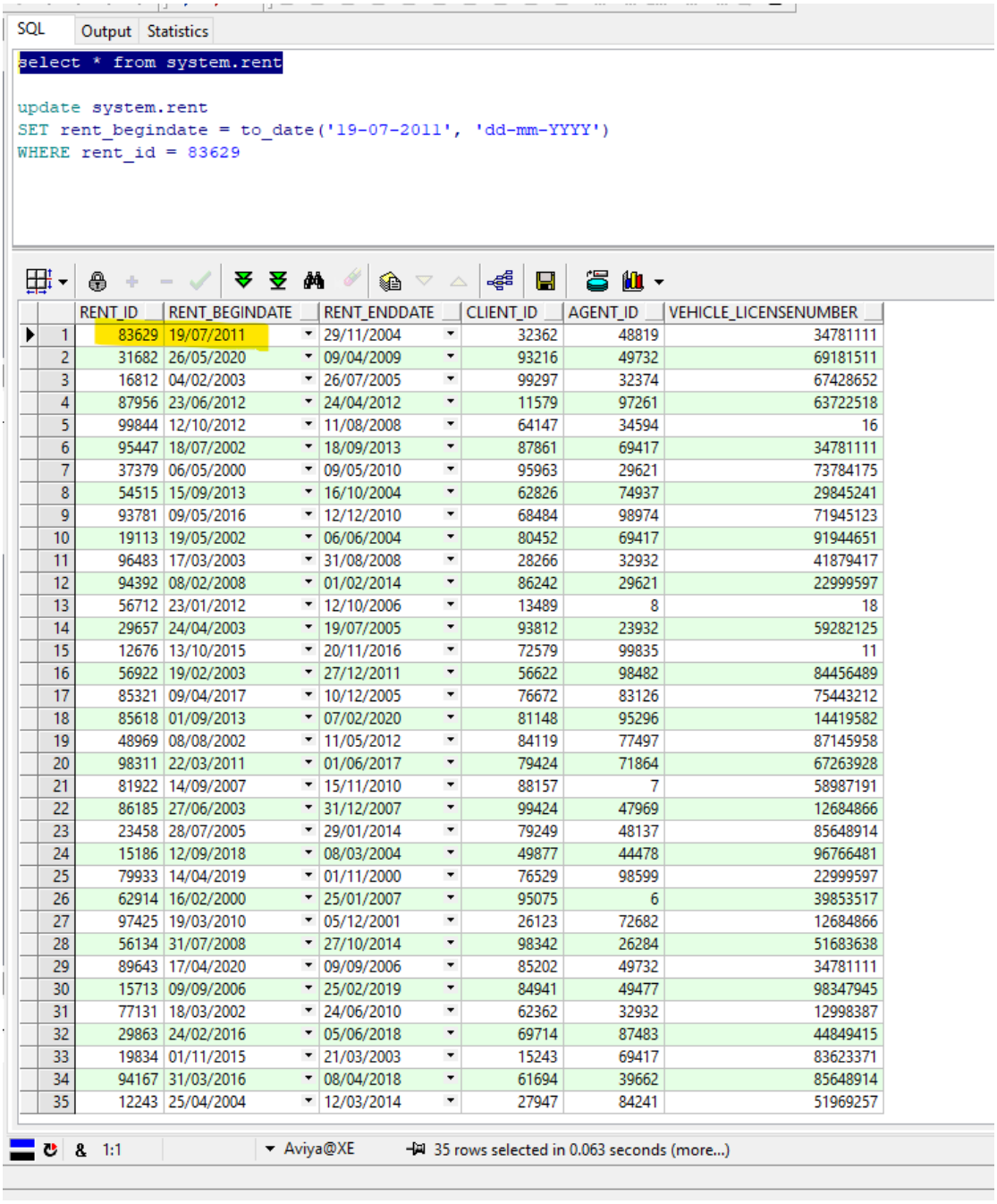


	RENT_ID	RENT_BEGINDATE	RENT_ENDDATE	CLIENT_ID	AGENT_ID	VEHICLE_LICENSENUMBER
1	83629	24/10/2009	29/11/2004	32362	48819	34781111
2	31682	26/05/2020	09/04/2009	93216	49732	69181511
3	16812	04/02/2003	26/07/2005	99297	32374	67428652
4	87956	23/06/2012	24/04/2012	11579	97261	63722518
5	99844	12/10/2012	11/08/2008	64147	34594	16
6	95447	18/07/2002	18/09/2013	87861	69417	34781111
7	37379	06/05/2000	09/05/2010	95963	29621	73784175
8	54515	15/09/2013	16/10/2004	62826	74937	29845241
9	93781	09/05/2016	12/12/2010	68484	98974	71945123
10	19113	19/05/2002	06/06/2004	80452	69417	91944651
11	96483	17/03/2003	31/08/2008	28266	32932	41879417
12	94392	08/02/2008	01/02/2014	86242	29621	22999597
13	56712	23/01/2012	12/10/2006	13489	8	18
14	29657	24/04/2003	19/07/2005	93812	23932	59282125
15	12676	13/10/2015	20/11/2016	72579	99835	11
16	56922	19/02/2003	27/12/2011	56622	98482	84456489
17	85321	09/04/2017	10/12/2005	76672	83126	75443212
18	85618	01/09/2013	07/02/2020	81148	95296	14419582
19	48969	08/08/2002	11/05/2012	84119	77497	87145958
20	98311	22/03/2011	01/06/2017	79424	71864	67263928
21	81922	14/09/2007	15/11/2010	88157	7	58987191
22	86185	27/06/2003	31/12/2007	99424	47969	12684866
23	23458	28/07/2005	29/01/2014	79249	48137	85648914
24	15186	12/09/2018	08/03/2004	49877	44478	96766481
25	79933	14/04/2019	01/11/2000	76529	98599	22999597
26	62914	16/02/2000	25/01/2007	95075	6	39853517
27	97425	19/03/2010	05/12/2001	26123	72682	12684866
28	56134	31/07/2008	27/10/2014	98342	26284	51683638
29	89643	17/04/2020	09/09/2006	85202	49732	34781111
30	15713	09/09/2006	25/02/2019	84941	49477	98347945

קוד העדכון:

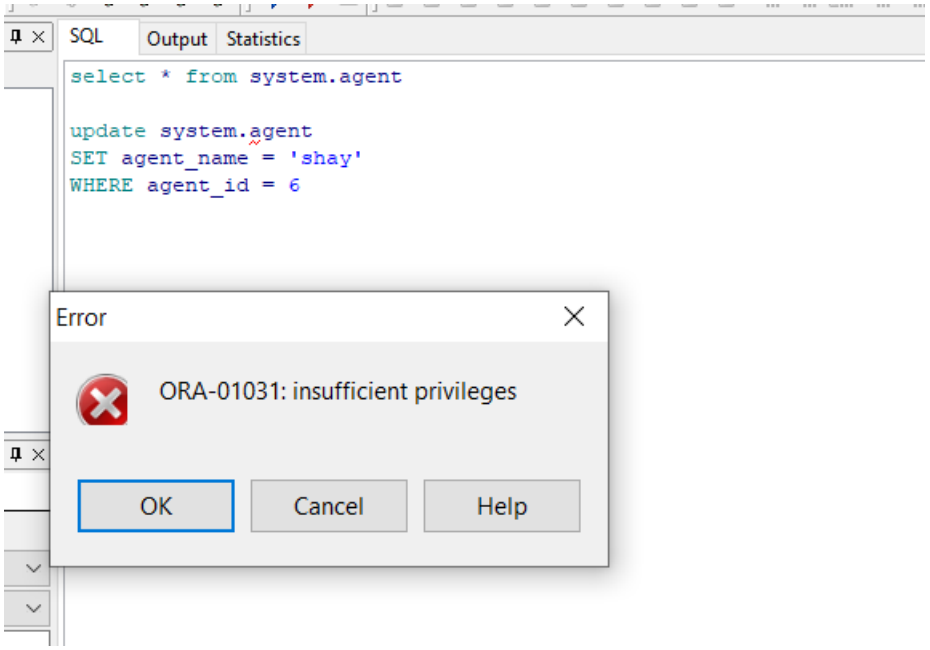
```
update system.rent
SET rent_begindate = to_date('19-07-2011', 'dd-mm-YYYY')
WHERE rent_id = 83629
```

נראה שהעדכון אכן בוצע:



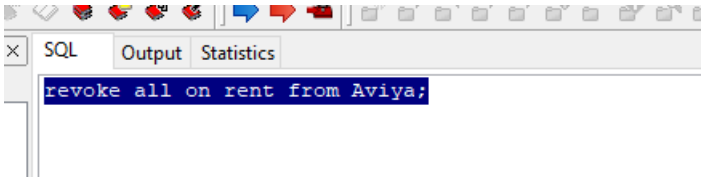
	RENT_ID	RENT_BEGINDATE	RENT_ENDDATE	CLIENT_ID	AGENT_ID	VEHICLE_LICENSENUMBER
1	83629	19/07/2011	29/11/2004	32362	48819	34781111
2	31682	26/05/2020	09/04/2009	93216	49732	69181511
3	16812	04/02/2003	26/07/2005	99297	32374	67428652
4	87956	23/06/2012	24/04/2012	11579	97261	63722518
5	99844	12/10/2012	11/08/2008	64147	34594	16
6	95447	18/07/2002	18/09/2013	87861	69417	34781111
7	37379	06/05/2000	09/05/2010	95963	29621	73784175
8	54515	15/09/2013	16/10/2004	62826	74937	29845241
9	93781	09/05/2016	12/12/2010	68484	98974	71945123
10	19113	19/05/2002	06/06/2004	80452	69417	91944651
11	96483	17/03/2003	31/08/2008	28266	32932	41879417
12	94392	08/02/2008	01/02/2014	86242	29621	22999597
13	56712	23/01/2012	12/10/2006	13489	8	18
14	29657	24/04/2003	19/07/2005	93812	23932	59282125
15	12676	13/10/2015	20/11/2016	72579	99835	11
16	56922	19/02/2003	27/12/2011	56622	98482	84456489
17	85321	09/04/2017	10/12/2005	76672	83126	75443212
18	85618	01/09/2013	07/02/2020	81148	95296	14419582
19	48969	08/08/2002	11/05/2012	84119	77497	87145958
20	98311	22/03/2011	01/06/2017	79424	71864	67263928
21	81922	14/09/2007	15/11/2010	88157	7	58987191
22	86185	27/06/2003	31/12/2007	99424	47969	12684866
23	23458	28/07/2005	29/01/2014	79249	48137	85648914
24	15186	12/09/2018	08/03/2004	49877	44478	96766481
25	79933	14/04/2019	01/11/2000	76529	98599	22999597
26	62914	16/02/2000	25/01/2007	95075	6	39853517
27	97425	19/03/2010	05/12/2001	26123	72682	12684866
28	56134	31/07/2008	27/10/2014	98342	26284	51683638
29	89643	17/04/2020	09/09/2006	85202	49732	34781111
30	15713	09/09/2006	25/02/2019	84941	49477	98347945
31	77131	18/03/2002	24/06/2010	62362	32932	12998387
32	29863	24/02/2016	05/06/2018	69714	87483	44849415
33	19834	01/11/2015	21/03/2003	15243	69417	83623371
34	94167	31/03/2016	08/04/2018	61694	39662	85648914
35	12243	25/04/2004	12/03/2014	27947	84241	51969257

ננסה לבצע פעולה עבורה לא ניתנה הרשאת גישה על הטבלה:

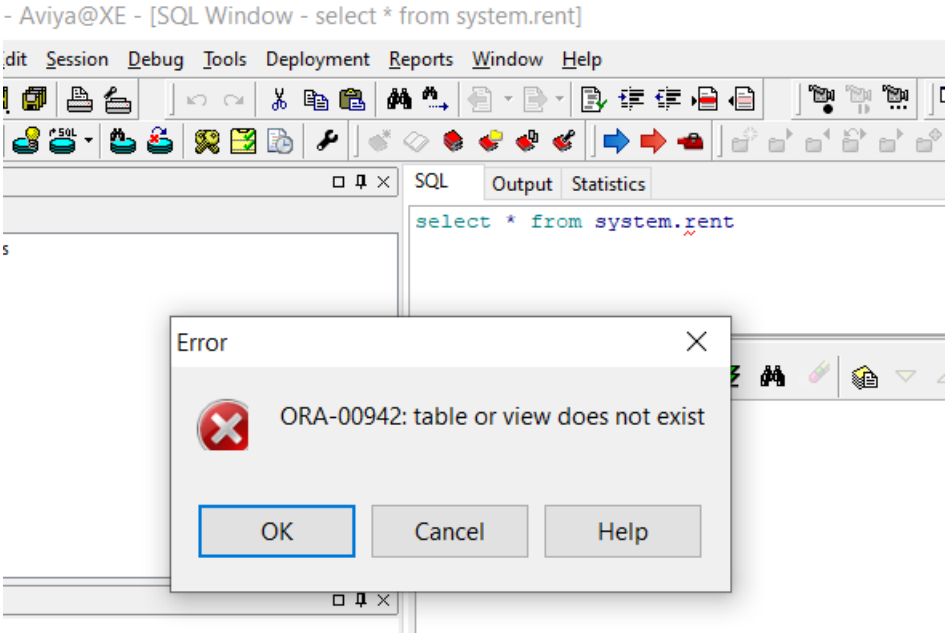


הפעולה אכן בלתי אפשרית עבור המשתמש אביה.

נבצע revoke:



ואכן ההרשאה בוטלה:



שלב 4:

מטלה 8:

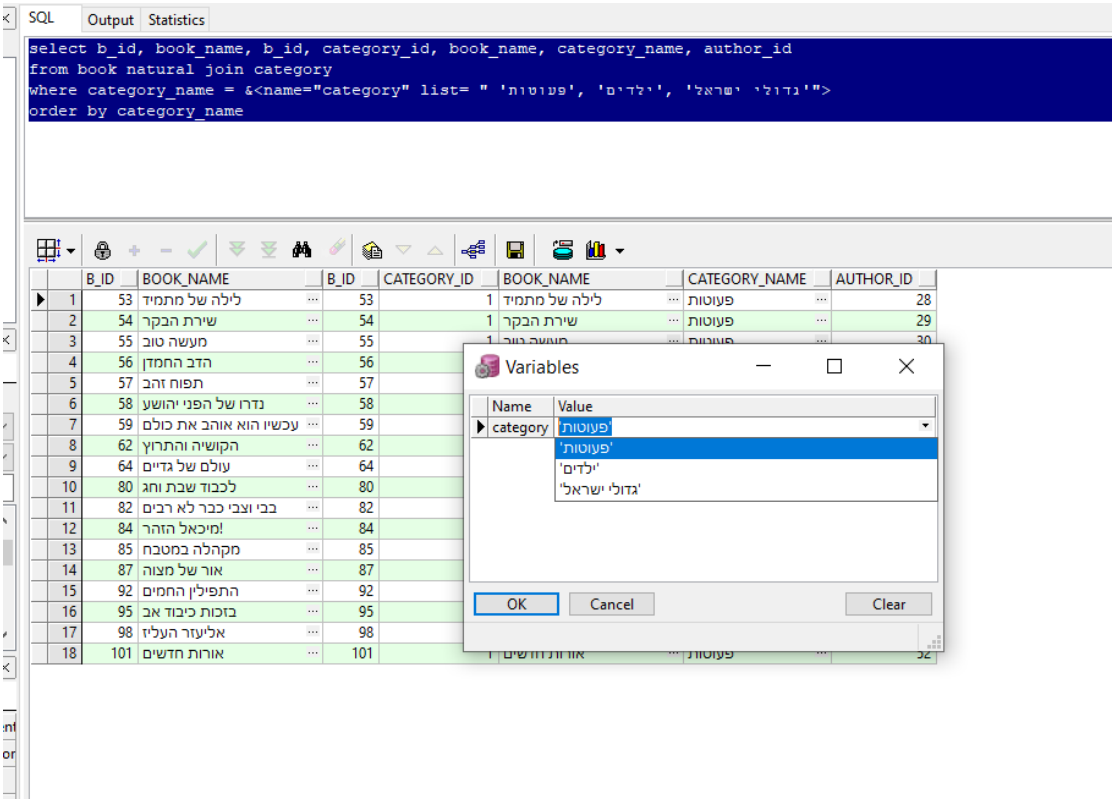
שאילתות עם פרמטרים:

השאילתה:

נרצה לקבל פרטים על הספרים מקטגורה מסוימת בלבד:

```
select b_id, book_name, b_id, category_id, book_name, category_name,
author_id
from book natural join category
where category_name = &<name="category" list= " ' ,'ילדים'' ,'פעוטות'
גדולי ישראל''">
order by category_name
```

התבקשנו לבחור מרשימת הקטגוריות:



ואכן, בחרנו ב'פעוטות' וקיבלנו:

SQL Output Statistics

```
select b_id, book_name, b_id, category_id, book_name, category_name, author_id
from book natural join category
where category_name = &<name="category" list= " 'פעוטות', 'ילדים', 'נדולי ישראל'">
order by category_name
```

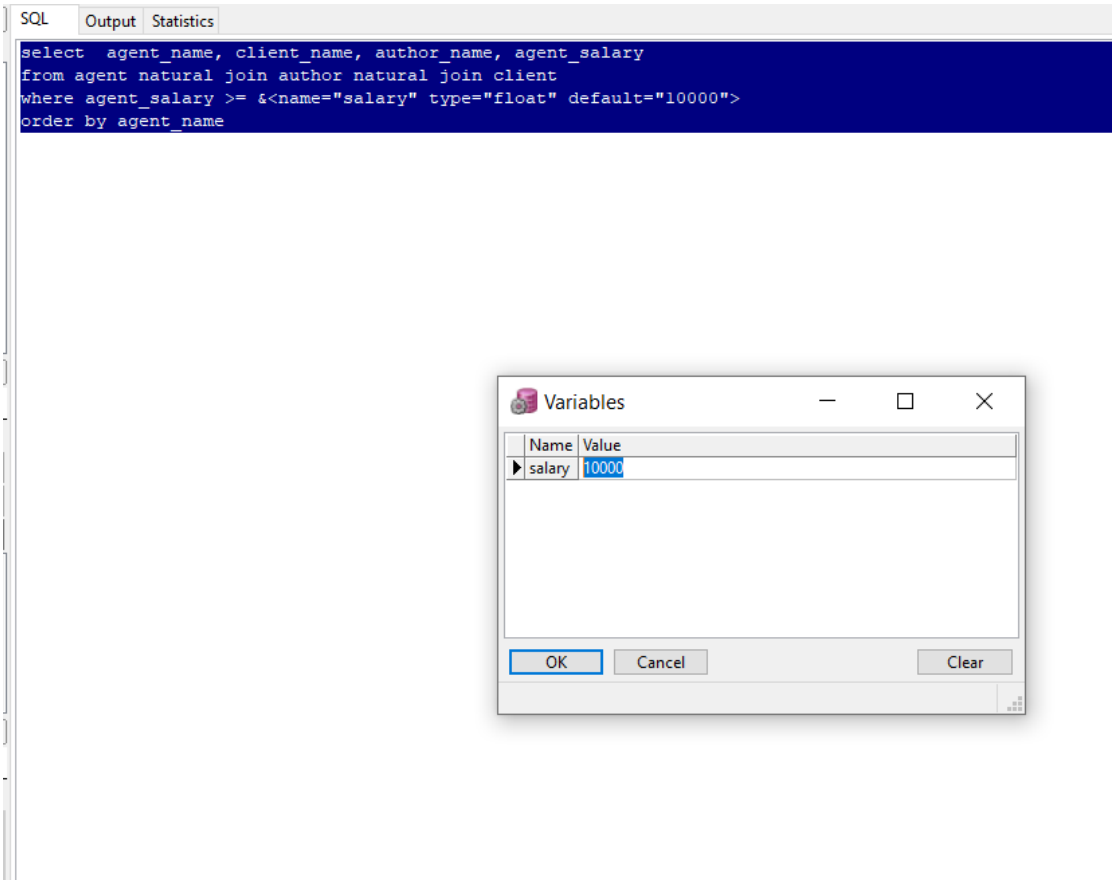
	B_ID	BOOK_NAME	B_ID	CATEGORY_ID	BOOK_NAME	CATEGORY_NAME	AUTHOR_ID
1	53	לילה של מתמיד	53	1	לילה של מתמיד	פעוטות	28
2	54	שירת הבקר	54	1	שירת הבקר	פעוטות	29
3	55	מעשה טוב	55	1	מעשה טוב	פעוטות	30
4	56	הדב החמדן	56	1	הדב החמדן	פעוטות	31
5	57	תפוח זהב	57	1	תפוח זהב	פעוטות	32
6	58	נדרו של הפני יהושע	58	1	נדרו של הפני יהושע	פעוטות	28
7	59	עכשיו הוא אוהב את כולם	59	1	עכשיו הוא אוהב את כולם	פעוטות	33
8	62	הקושיה והתרוץ	62	1	הקושיה והתרוץ	פעוטות	35
9	64	עולם של גדיים	64	1	עולם של גדיים	פעוטות	36
10	80	לכבוד שבת וחג	80	1	לכבוד שבת וחג	פעוטות	44
11	82	בבי וצבי כבר לא רבים	82	1	בבי וצבי כבר לא רבים	פעוטות	46
12	84	מיכאל הזהר!	84	1	מיכאל הזהר!	פעוטות	47
13	85	מקהלה במטבח	85	1	מקהלה במטבח	פעוטות	48
14	87	אור של מצוה	87	1	אור של מצוה	פעוטות	50
15	92	התפילין החמים	92	1	התפילין החמים	פעוטות	52
16	95	בזכות כיבוד אב	95	1	בזכות כיבוד אב	פעוטות	55
17	98	אליעזר העליז	98	1	אליעזר העליז	פעוטות	57
18	101	אורות חדשים	101	1	אורות חדשים	פעוטות	52

שאילתה נוספת עם פרמטר:

נרצה לקבל את שמות הסוכנים והלקוחות שביצעו השאלת ספר ואת שם הסופר, אך רק לסוכנים שמשכורתם גדולה מסכום מסויים ולכן נוסיף פרמטר עבור הסכום:

```
select  agent_name, client_name, author_name, agent_salary
from agent natural join author natural join client
where agent_salary >= &<name="salary" type="float" default="10000">
order by agent_name
```

הפרמטר הינו ערך המשכורת ולו ערך ברירת מחדל שערכו 10000:



השארנו את ערך ברירת המחדל ואכן קיבלנו תוצאה:

SQL | Output | Statistics

```
select  agent_name, client_name, author_name, agent_salary
from agent natural join author natural join client
where agent_salary >= &<name="salary" type="float" default="10000">
order by agent_name
```

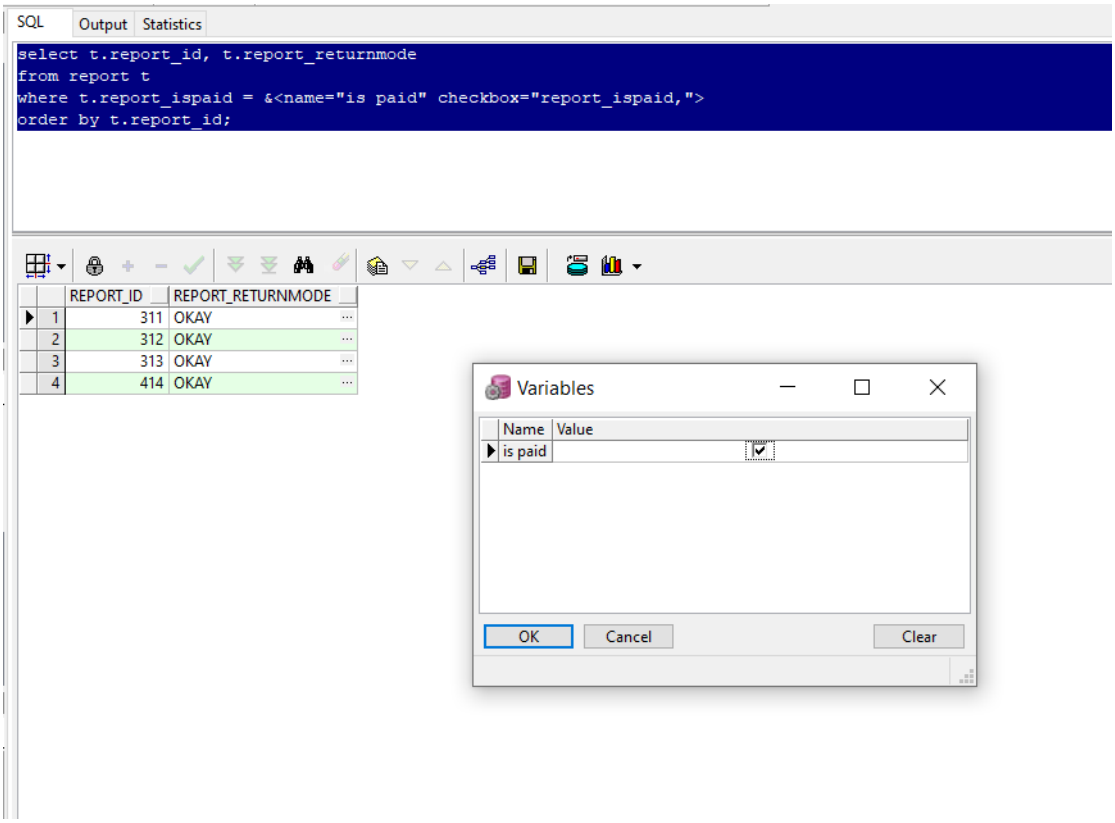
	AGENT_NAME	CLIENT_NAME	AUTHOR_NAME	AGENT_SALARY
1	bat sheva	Miko	''"'-'·'·'©''·'	13000.00
2	bat sheva	Suzi	''"'-'·'·'©''·'	13000.00
3	bat sheva	Fionnula	''"'-'·'·'©''·'	13000.00
4	bat sheva	Davy	''"'-'·'·'©''·'	13000.00
5	bat sheva	Lloyd	''"'-'·'·'©''·'	13000.00
6	bat sheva	Glenn	''"'-'·'·'©''·'	13000.00
7	bat sheva	Bonnie	''"'-'·'·'©''·'	13000.00
8	bat sheva	Stanley	''"'-'·'·'©''·'	13000.00
9	bat sheva	Clay	''"'-'·'·'©''·'	13000.00
10	bat sheva	Brendan	''"'-'·'·'©''·'	13000.00
11	bat sheva	Orlando	''"'-'·'·'©''·'	13000.00
12	bat sheva	Humberto	''"'-'·'·'©''·'	13000.00
13	bat sheva	Fats	''"'-'·'·'©''·'	13000.00
14	bat sheva	Emilio	''"'-'·'·'©''·'	13000.00
15	bat sheva	Bridget	''"'-'·'·'©''·'	13000.00
16	bat sheva	Malcolm	''"'-'·'·'©''·'	13000.00
17	bat sheva	Tobey	''"'-'·'·'©''·'	13000.00
18	bat sheva	Suzanne	''"'-'·'·'©''·'	13000.00
19	bat sheva	Ashton	''"'-'·'·'©''·'	13000.00
20	bat sheva	Embeth	''"'-'·'·'©''·'	13000.00
21	bat sheva	Rhett	''"'-'·'·'©''·'	13000.00
22	bat sheva	Kirk	''"'-'·'·'©''·'	13000.00
23	bat sheva	Geraldine	''"'-'·'·'©''·'	13000.00
24	bat sheva	Dom	''"'-'·'·'©''·'	13000.00
25	bat sheva	Jaime	''"'-'·'·'©''·'	13000.00
26	bat sheva	Roy	''"'-'·'·'©''·'	13000.00
27	bat sheva	Trick	''"'-'·'·'©''·'	13000.00
28	bat sheva	Ani	''"'-'·'·'©''·'	13000.00
29	bat sheva	Julie	''"'-'·'·'©''·'	13000.00
30	bat sheva	Latin	''"'-'·'·'©''·'	13000.00
31	bat sheva	Angelina	''"'-'·'·'©''·'	13000.00
32	bat sheva	Reese	''"'-'·'·'©''·'	13000.00
33	bat sheva	Dabney	''"'-'·'·'©''·'	13000.00
34	bat sheva	Lonnie	''"'-'·'·'©''·'	13000.00
35	bat sheva	Kiefer	''"'-'·'·'©''·'	13000.00

& 5:1 | SYSTEM@XE | 35 rows selected in 0.313 seconds (more...)

שאילתה נוספת:

נרצה לקבל את מספרי ההשכרות ואת מצב ההחזרה של הרכב ונוסיף צ'ק בוקס עבור האם שולם או לא:

```
select t.report_id, t.report_returnmode
from report t
where t.report_ispaid = &<name="is paid" checkbox="report_ispaid,">
order by t.report_id;
```



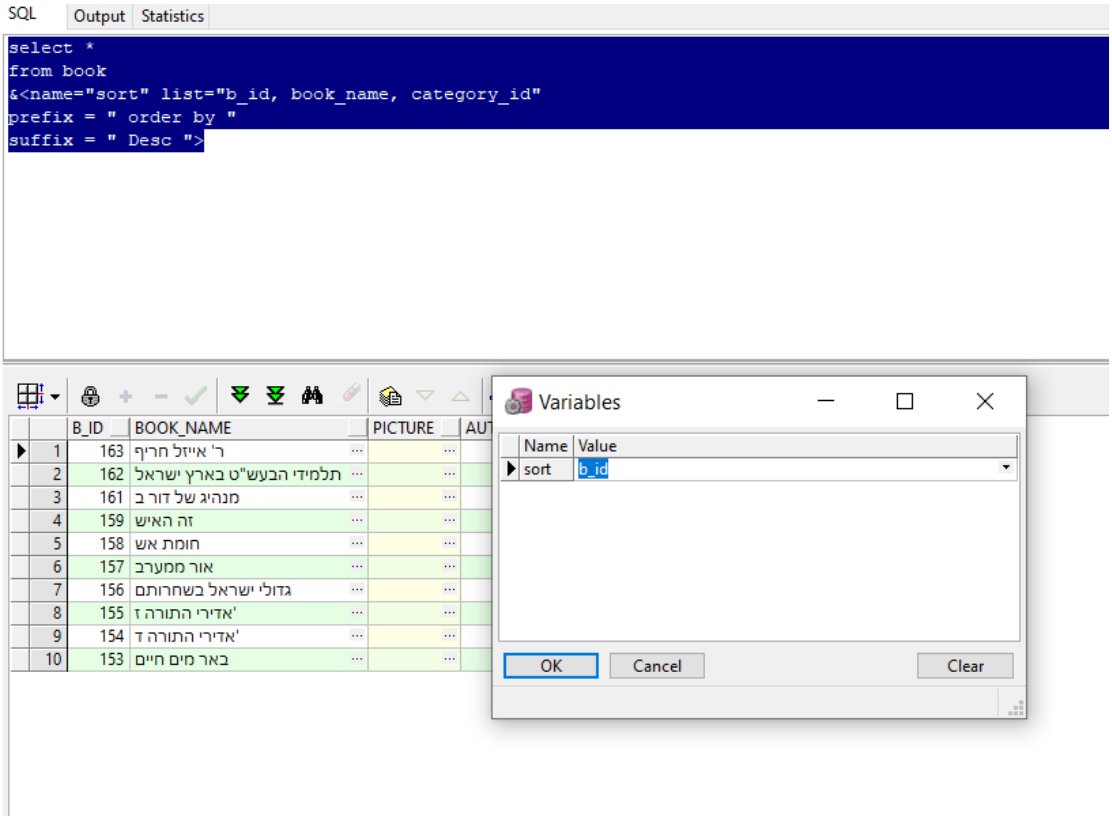
שאילתה נוספת:

נרצה לקבל את פרטי הספר ולמיין עפ"י עמודה מסויימת:

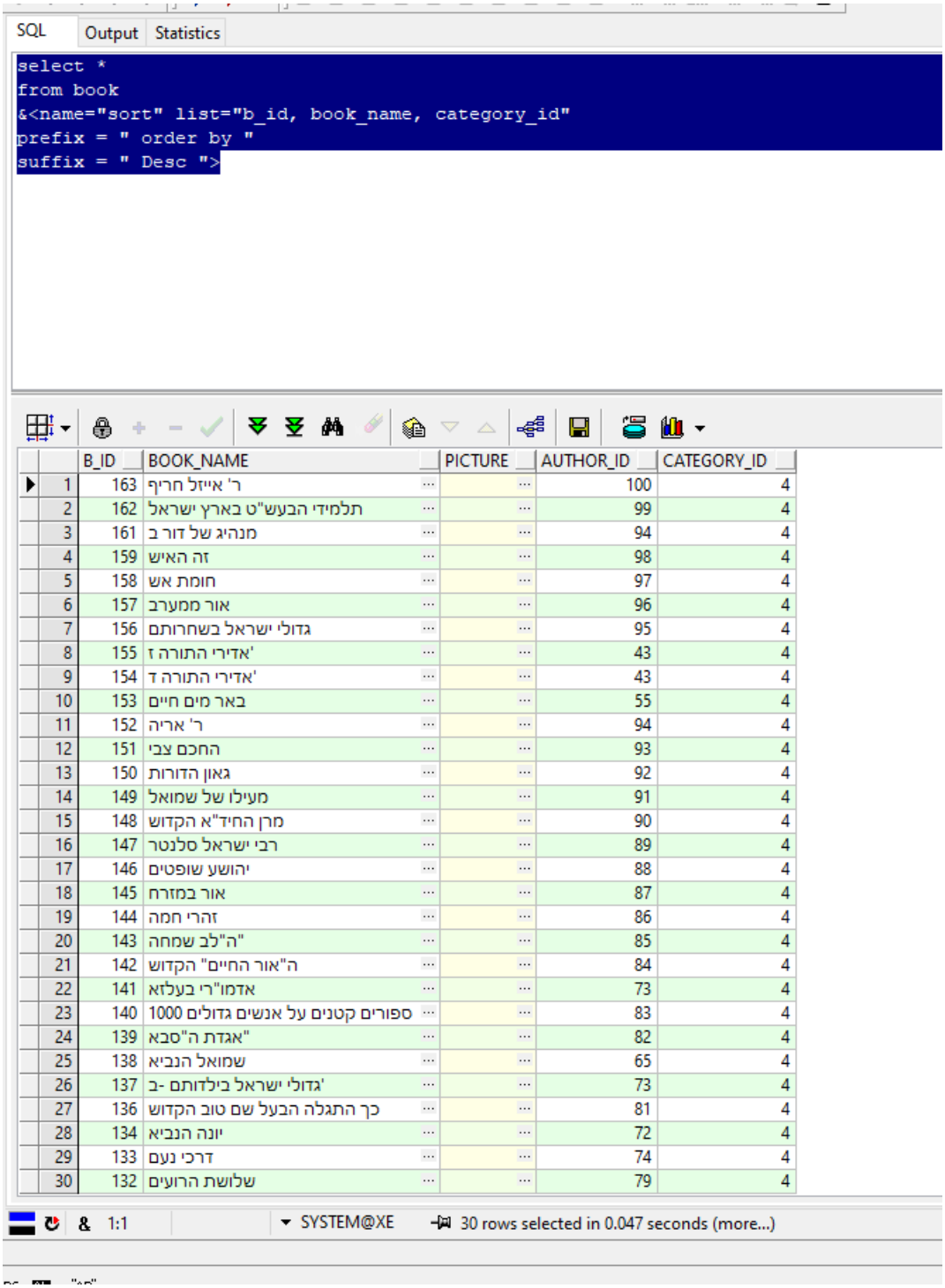
```
select *
from book
&<name="sort" list="b_id, book_name, category_id"
prefix = " order by "
suffix = " Desc ">
```

המשתמש בוחר ע"י פרמטר איזו עמודה הוא מעוניין למיין:

בחרנו למיין עפ"י הid של הספר:



ואכן קיבלנו את התוצאה ממוינת עפ"י id של הספר, בסדר יורד:

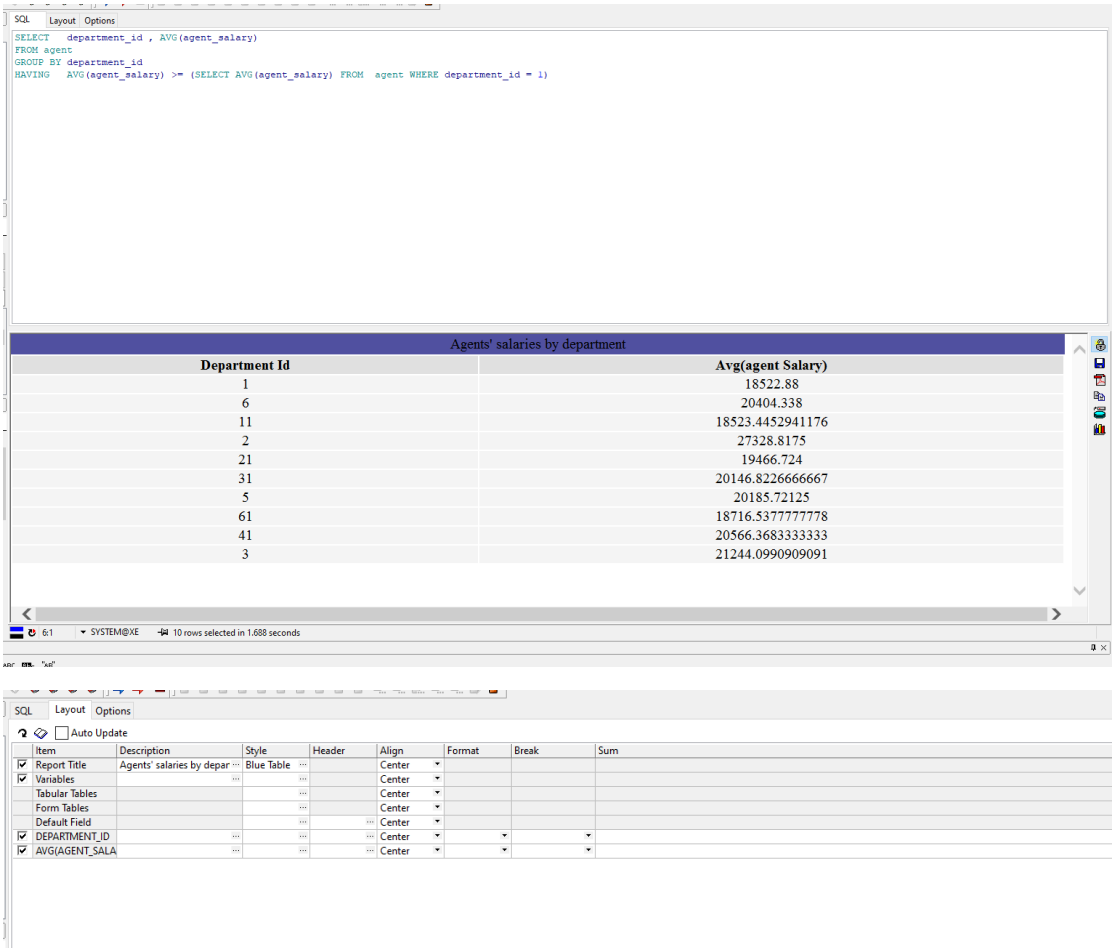


דוחות:

דוח:

נרצה את ממוצע המשכורות במחלקות בהן הממוצע גדול או שווה לזה של מחלקה 1:

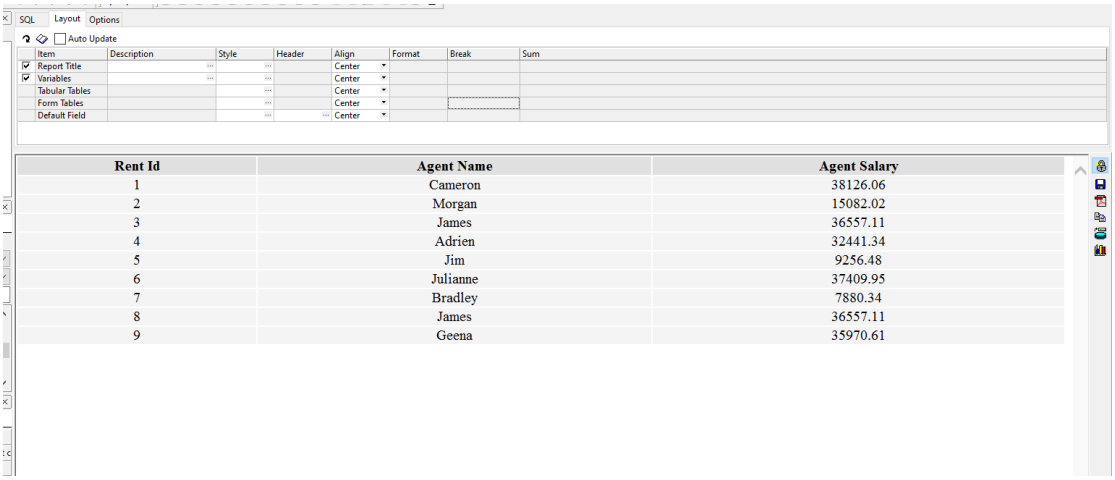
```
SELECT   department_id , AVG(agent_salary)
FROM agent
GROUP BY department_id
HAVING   AVG(agent_salary) >= (SELECT AVG(agent_salary) FROM  agent
WHERE department_id = 1)
```



דוח:

נרצה לראות דוח עם ההשכרות הראשונות שבוצעו בארגון, ומשכורותיהם של הסוכנים שביצעו אותן:

```
select rent_id, agent_name, agent_salary
from rent natural join agent
where rent_id < 10
order by rent_id
```

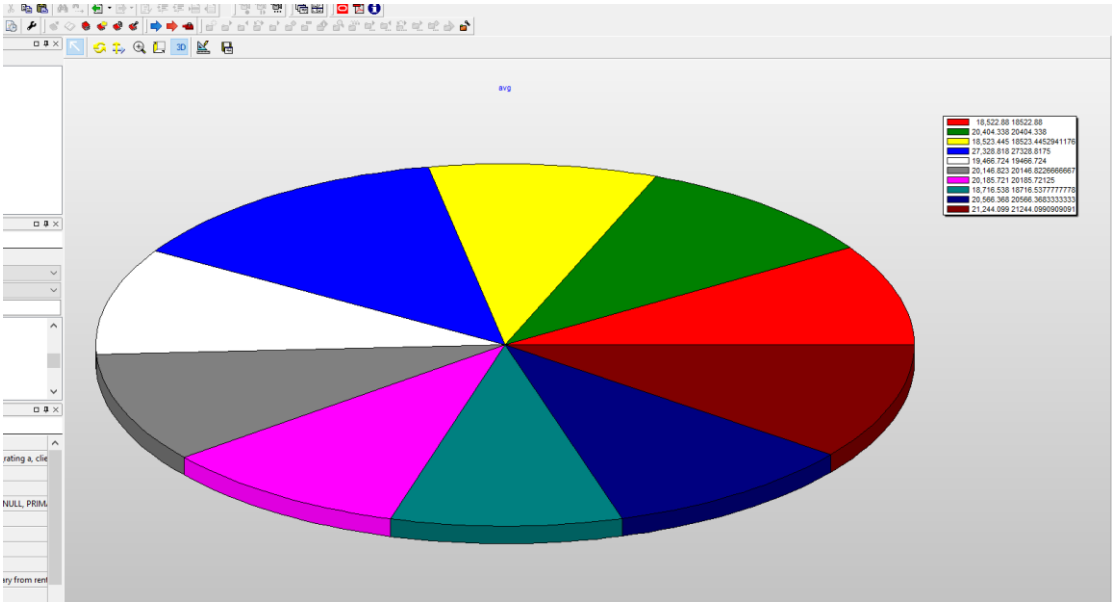


Rent Id	Agent Name	Agent Salary
1	Cameron	38126.06
2	Morgan	15082.02
3	James	36557.11
4	Adrien	32441.34
5	Jim	9256.48
6	Julianne	37409.95
7	Bradley	7880.34
8	James	36557.11
9	Geena	35970.61

גרף:

נרצה לראות גרף עם ממוצעי המשכורות במחלקות בהן הממוצע גדול או שווה לזה של מחלקה 1:

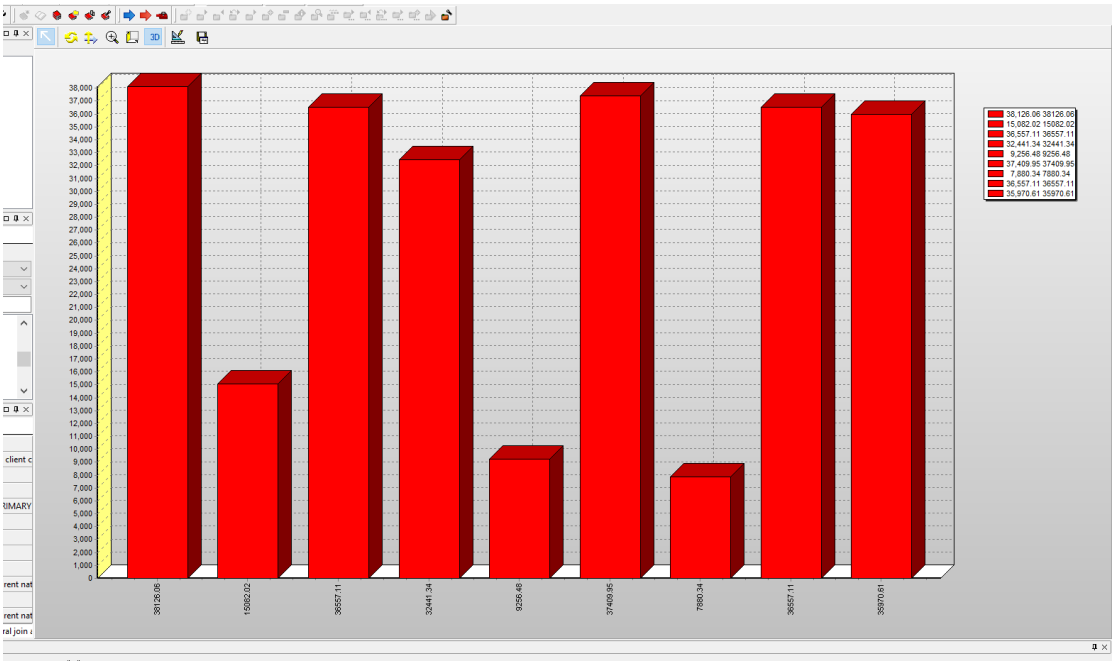
```
SELECT   department_id , AVG(agent_salary)
FROM agent
GROUP BY department_id
HAVING   AVG(agent_salary) >= (SELECT AVG(agent_salary) FROM  agent
WHERE department_id = 1)
```



גרף:

נרצה לראות בגרף את המשכורות של הסוכנים שביצעו את ההשכרות הראשונות שבוצעו בארגון:

```
select rent_id, agent_name, agent_salary
from rent natural join agent
where rent_id < 10
order by rent_id
```



שלב 4:

מטלה 9:

1. פרוצדורות:

1.1 הדפסת הסוכן /ים שסגר הכי הרבה עסקאות:

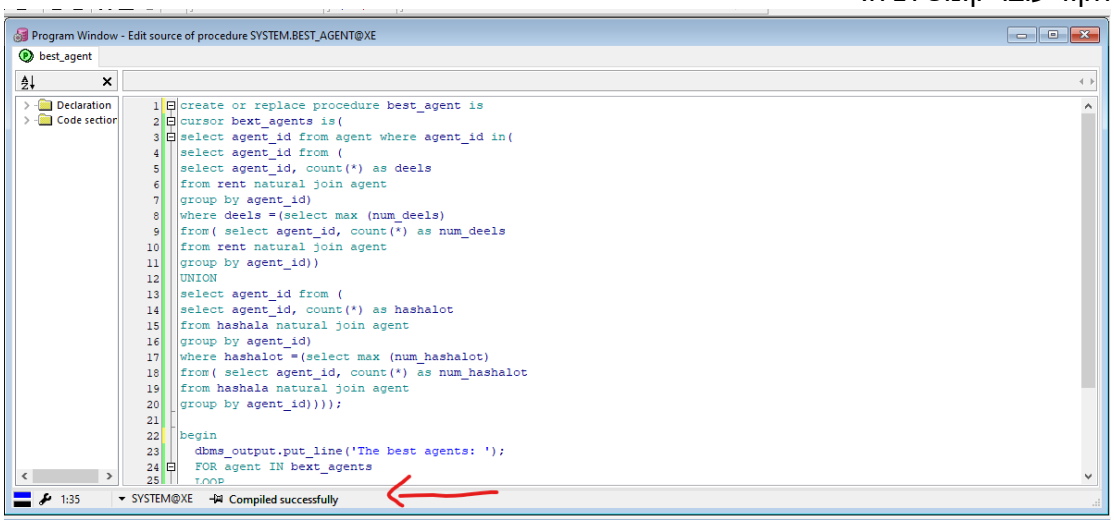
הפרוצדורה מדפיסה את הסוכן הכי טוב שביצע הכי הרבה עסקאות רכב או שביצע הכי הרבה השאלות ספרים. הפרוצדורה לא לוקחת בחשבון סוכן שביצע גם השכרות רכב וגם השאלות ספרים בחישוב הסוכן המוצלח.

הקוד:

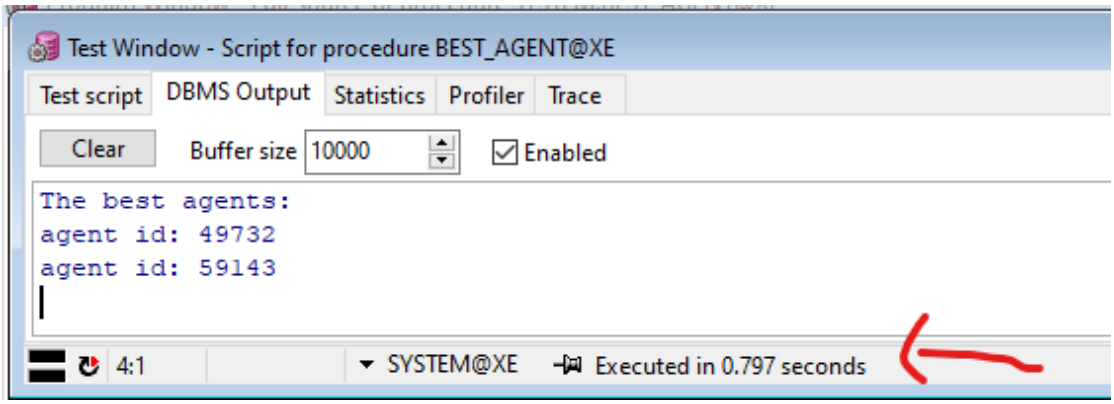
```
create or replace procedure best_agent is
cursor bext_agents is(
select agent_id from agent where agent_id in(
select agent_id from (
select agent_id, count(*) as deels
from rent natural join agent
group by agent_id)
where deels =(select max (num_deels)
from( select agent_id, count(*) as num_deels
from rent natural join agent
group by agent_id))
```

```
UNION
select agent_id from (
select agent_id, count(*) as hashalot
from hashala natural join agent
group by agent_id)
where hashalot =(select max (num_hashalot)
from( select agent_id, count(*) as num_hashalot
from hashala natural join agent
group by agent_id))));

begin
  dbms_output.put_line('The best agents: ');
  FOR agent IN bext_agents
  LOOP
  dbms_output.put_line('agent id: '|| agent.agent_id);
  END LOOP;

end;
```

הקוד עובר קומפילציה:



הפלט:

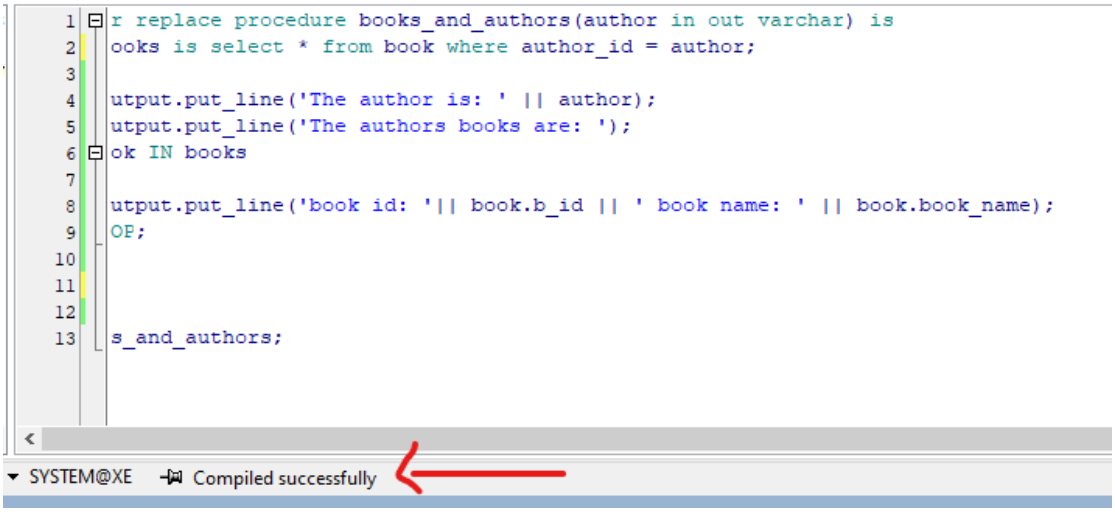


1.2 הפרוצדורה מקבלת כקלט author_id ומדפיסה את כל הספרים שהוא חיבר.

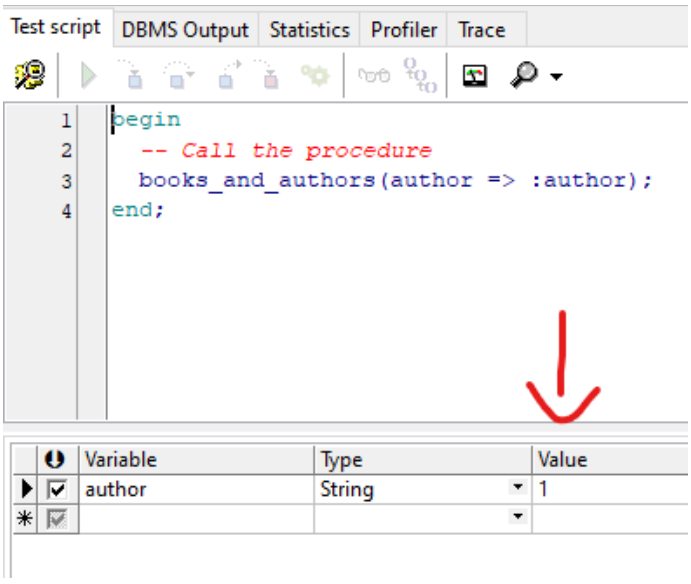
הקוד:

```
create or replace procedure books_and_authors(author in out varchar)
is
cursor books is select * from book where author_id = author;
begin
  dbms_output.put_line('The author is: ' || author);
  dbms_output.put_line('The authors books are: ');
  FOR book IN books
  LOOP
  dbms_output.put_line('book id: '|| book.b_id || ' book name: ' ||
book.book_name);
  END LOOP;



end books_and_authors;
```

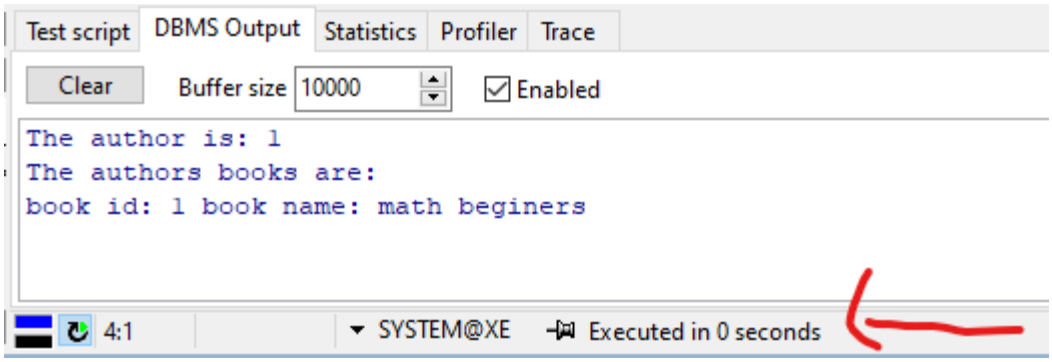
הקוד עובר קומפילציה:



הקלט:



הפלט:



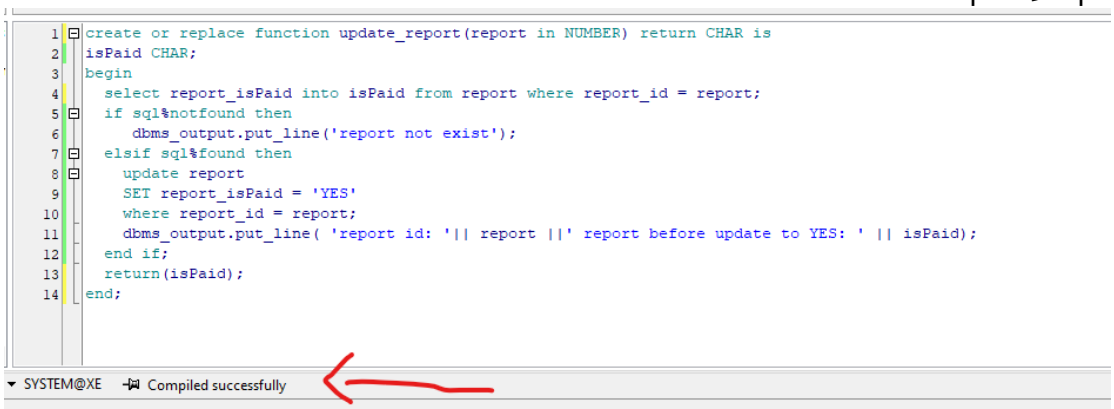
2. פונקציות

2.1 הפונקציה מקבלת report_id, מחזירה את הreport_isPaid שלו, ומשנה את report_isPaid שלו ל'YES' .

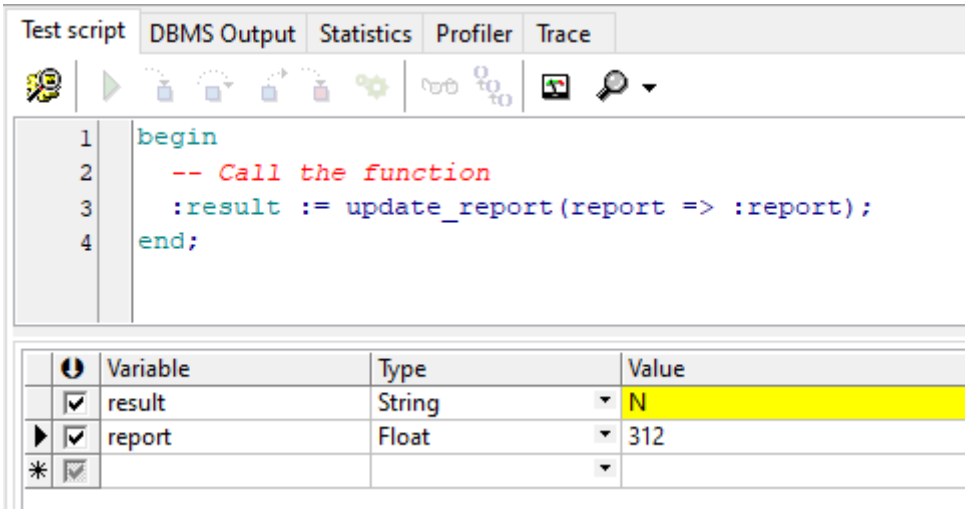
הקוד:

```
create or replace function update_report(report in NUMBER) return
CHAR is
isPaid CHAR;
begin
  select report_isPaid into isPaid from report where report_id =
report;
  if sql%notfound then
     dbms_output.put_line('report not exist');
  elsif sql%found then
    update report
    SET report_isPaid = 'YES'
    where report_id = report;
    dbms_output.put_line( 'report id: '|| report ||' report before
update to YES: ' || isPaid);
  end if;
  return(isPaid);
end;
```

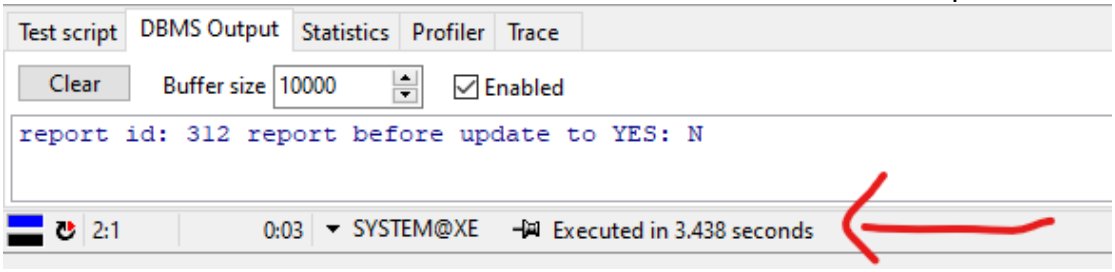
הקוד עובר קומפילציה:



הקלט והreturn value של התוכנית:



ההדפסות וזמן הריצה:



2.2 הפונקציה מקבלת vehicle_id והנחה ומחזירה את המחיר הרכב לאחר ההנחה.
הקוד:

```
create or replace function calculates_discount(licensenumber in
number, discount in number) return number is
  price number;
  vehicle_model model.model_id%type;

begin
 select model_id into vehicle_model from vehicle where
vehicle_licensenumber = licensenumber;
 update model
    SET model_price = model_price - (model_price * discount / 100)
    where model_id = vehicle_model;
  select model_price into price from model where model_id =
vehicle_model;

  return(price);
end;
```

הקוד עובר קומפילציה:

```
create or replace function calculates_discount(licensenumber in number, discount in number) return number is
  price number;
  vehicle_model model.model_id%type;

begin
 select model_id into vehicle_model from vehicle where vehicle_licensenumber = licensenumber;
 update model
    SET model_price = model_price - (model_price * discount / 100)
    where model_id = vehicle_model;
  select model_price into price from model where model_id = vehicle_model;

  return(price);
end;
```

SYSTEM@XE Compiled successfully

הקלט והreturn value:

Test script | DBMS Output | Statistics | Profiler | Trace

```
begin
  -- Call the function
  :result := calculates_discount(licensenumber => :licensenumber,
                                 discount => :discount);
end;
```

Variable	Type	Value
result	Float	6400
licensenumber	Float	111
discount	Float	20

זמן הריצה:

Test script | DBMS Output | Statistics | Profiler | Trace

Clear | Buffer size 10000 | Enabled

1:1 0:01 SYSTEM@XE Executed in 1.453 seconds

3. טריגרים:

3.1 טריגר שמדפיס שהיה עדכון בכל פעם שמעדכנים את הareaid בטבלה city.

יצירת הטריגר:

Template Wizard

Trigger

Name: print_update_city

Fires: after

Event: update

Table or view: city

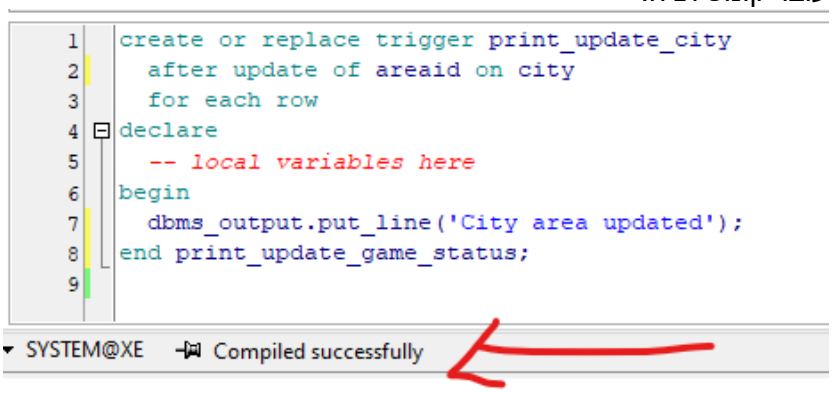
Statement level? ☐

OK | Cancel

הקוד:

```
create or replace trigger print_update_city
  after update of areaid on city
  for each row
declare
  -- local variables here
begin
  dbms_output.put_line('City area updated');
end print_update_game_status;
```

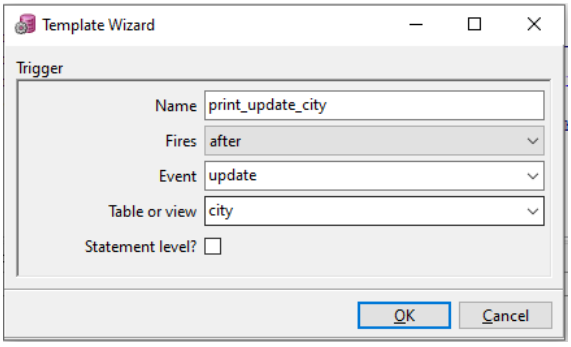
הקוד עובר קומפילציה:



...

3.2 טריגר שמדפיס בכל פעם שסוכן נוסף בהצלחה לטבלת agent.

יצירת הטריגר:



הקוד:

```
create or replace trigger insert_agent
  after insert on agent
  for each row
declare

begin
  dbms_output.put_line(:new.agent_id || ' successfully added');
end insert_agent;
```

הקוד עובר קומפילציה:

```
1  create or replace trigger insert_agent
2    after insert on agent
3    for each row
4  declare
5
6  begin
7    dbms_output.put_line(:new.agent_id || ' successfully added');
8  end insert_agent;
```

SYSTEM@XE Compiled successfully