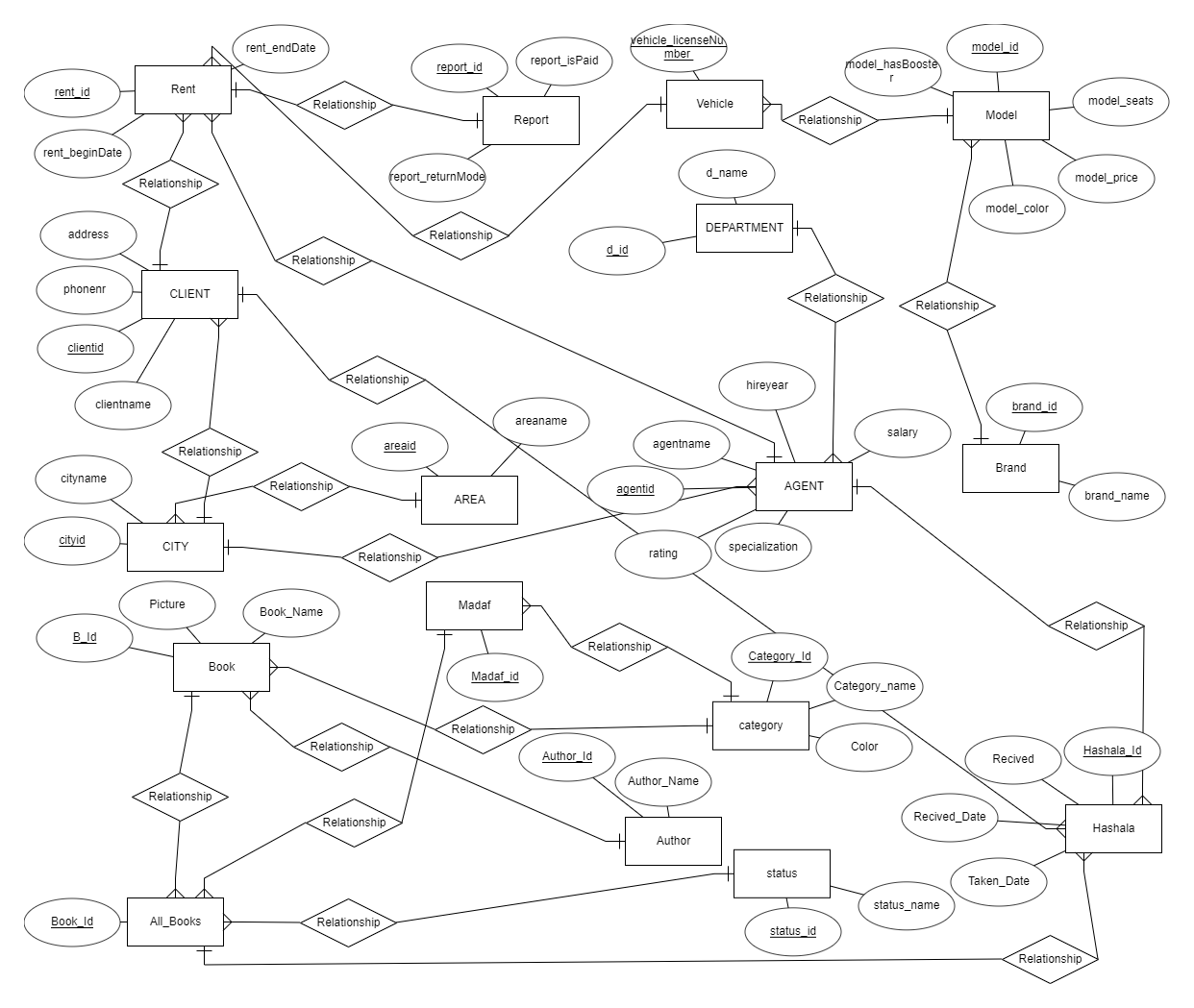
שלב 2:

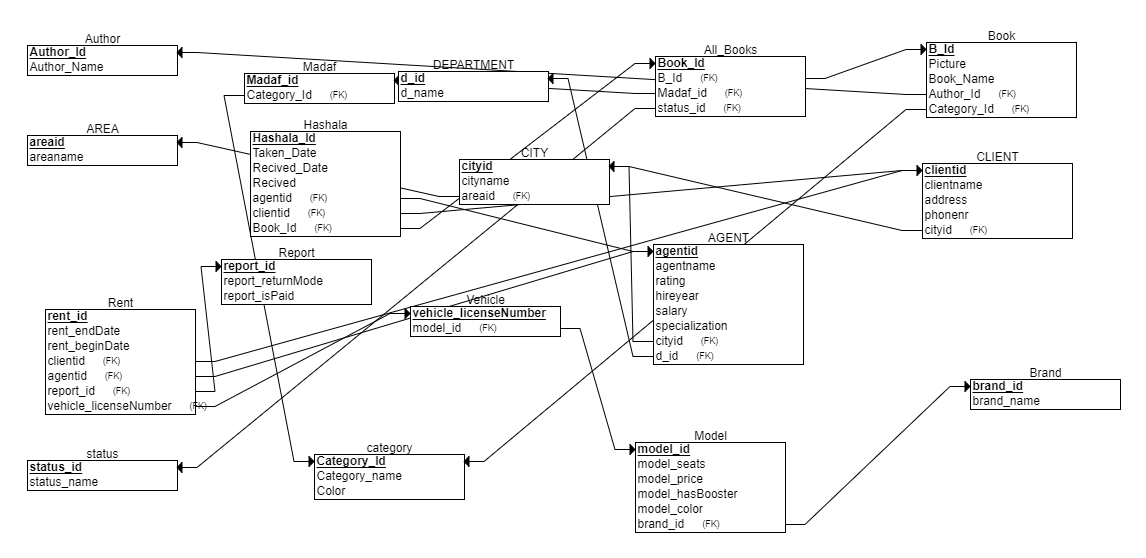
מטלה 5:

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.





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)

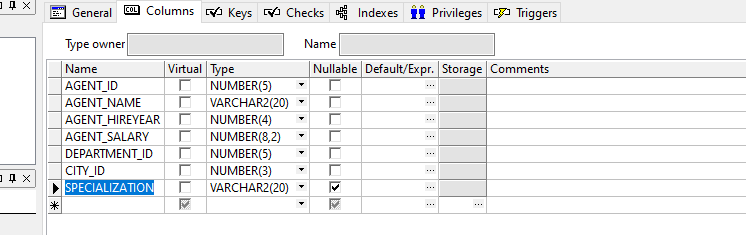
select s.areaid, s.areaname

from area s

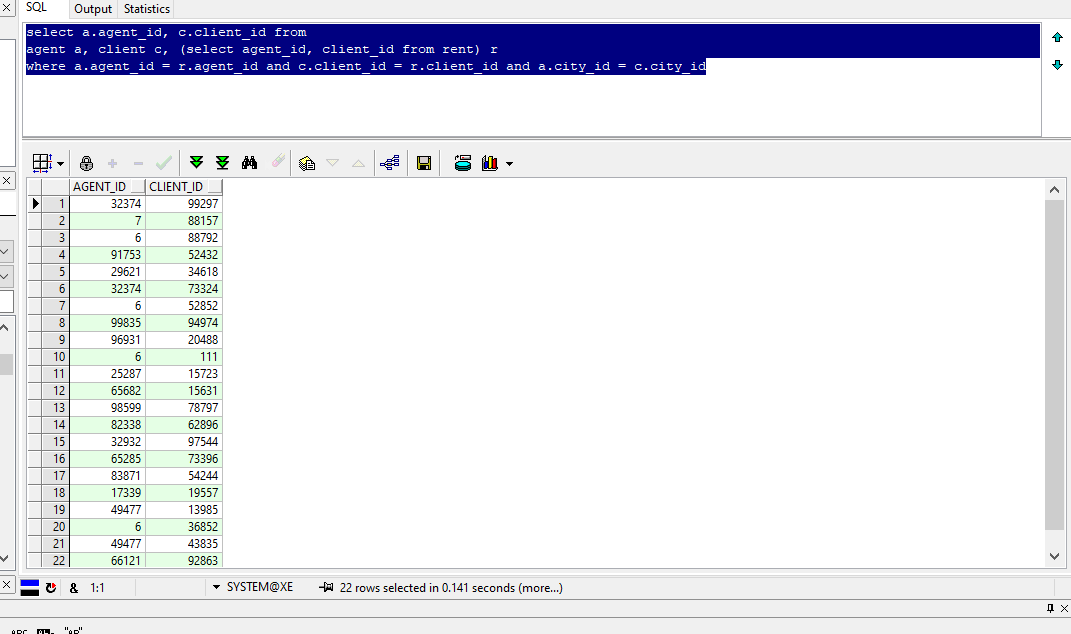
where s.areaid not in (select area\_id from area1);

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

נוסיף את התכונה התמחות לישות סוכן:

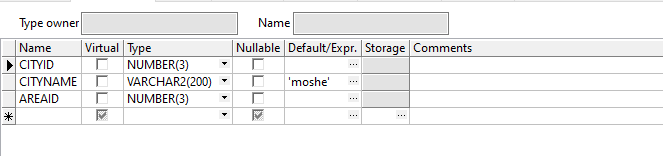


השאילתות אכן פועלות גם לאחר האינטגרציה:



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' );