**בס"ד**

**מיני פרויקט בבסיסי נתונים**

# עבודת הכנה והכרת התוכנה

## erdplus-diagramתרשים ERD

**תיאור הישויות והקשרים**

### ישויות

* Client - מאופיין ב- ת.ז של הלקוח, שם, כתובת, מספר פלאפון, עיר מגורים ומספר סוכן.
* Agent - מאופיין ב- ת.ז של הסוכן, שם סוכן, מספר אזור, דירוג, ותק, ת.ז של הבוס, משכורת.
* Area - מאופיין ב- מספר אזור ובשם האזור.
* City מאופיין ב- מספר אזור ובשם העיר.
* Schedule מאופיין ב- ת.ז של הסוכן, ת.ז של הלקוח ובזמן הפגישה.

### קשרים

* לכל סוכן- יש אזור עבודה יחידי, יכולים להיות הרבה לקוחות ויכולים להיות לו הרבה פגישות עם לקוחות.
* לכל אזור- יכולות להיות כמה ערים ויכולים להיות הרבה סוכנים שעובדים באזור זה.
* לכל עיר- יכולים להיות הרבה לקוחות שגרים בה והיא יכולה להיות באזור יחידי.
* לכל לוח זמנים- חייב להיות סוכן יחיד ולקוח יחיד הנפגשים בזמן מסוים.
* לכל לקוח-יש סוכן יחיד והוא גר בעיר יחידה ויכול להיות לו הרבה פגישות עם הסוכן שלו.

### נרמול הטבלאות

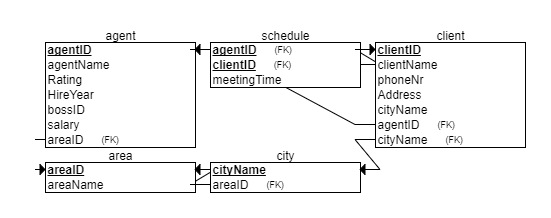
* Schedule (agentID, clientID, meetingTime)
* Agent (agentID, rating, hireYear, boosID, salary, areaID)
* Client (clientID, clientName, address, phoneNr, cityName)
* CityName (cityName, areaID)
* Area (areaID, areaName)

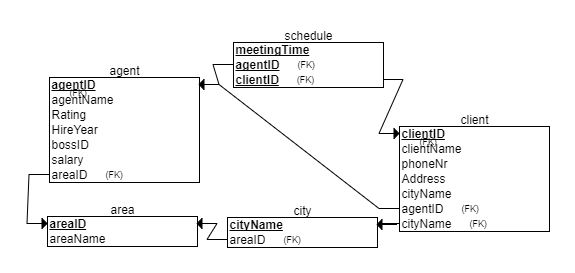
### פרוקים

היחסים עומדים ב- NF3 וב- BCNF : מכיוון שבכל טבלה, התלויות הפונקציונאליות הלא-טריוויאליות הן מהמפתח אל תכונות נוספות לכן מתקיים שלכל X Y , X הוא מפתח ולכן הם עומד בתנאים.

## תרשים DSD

כאן ניתן לראות את ה- DSD שהפקנו מתרשים ה- ERD שיצרנו.

אך זה לא כמו מה שמומש ב- create table שניתן לנו כי שם ב- schedule מופיע גם ה- meetingTime כ- praimery key ופה הוא לא מופיע.

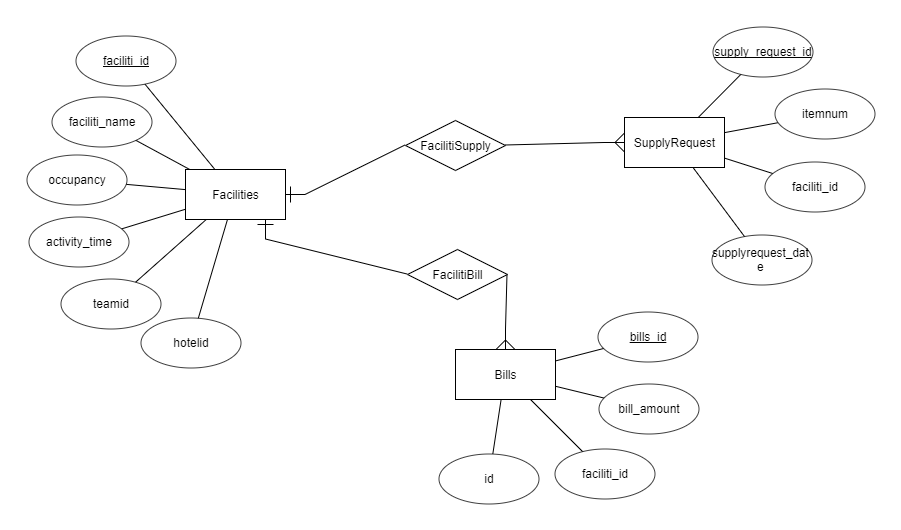
לכן נוסיף זאת לתרשים שלנו באופן ידני, ונקבל את התרשים הבא.

# הפרויקט שלנו

*אנו היינו אחראיים על 3 ישויות בפרוייקט שלנו סרטים אולמות וכרטיסים*

## תרשים ERD

כפי שהזכרנו, במחלקה שלנו, ישנן 3 ישויות: סרטים אולמות כרטיסים. בשלב הראשון יצרנו תרשים ERD שיתאר את הקשרים בין הישויות הללו ואת התכונות שלהן.



## תיאור הישויות והקשרים

### ישויות

* Tickets –יישות זאת אחראית על הכרטיסים לסרטים יישות זאת הינה חלשה מאחר והיא תלוייה ב באולמות וסרטים..
* ticketLine – מספר שורה של הכסא
* ticketchair – מספר כסא
* movieId – מזה סרט (FK)
* ticketStatus – האם היה שימוש בכרטיס או לא
* theaterId– מספר מזהה של התיאטרון בו יתקיים הסרט (FK)
* ticketId– מספר של הכרטיס (PK)
* Movies – יישות זאת אחראית על כל המידע על הסרטים אשר ב מסד נתונים .יישות זאת הינה חזזקה מאחר והיא לא תלויה בשום דבר אחר..
* movieId – מספר מזהה של הסרט (PK)
* releaseYear– שנת יציאה לאור של הסרט
* releaseDate- תאריך יציאה לאור של הסרט
* rating-דירוג הסרט ע"י הצופים
* Genre- ז'אנר (סוג ) הסרט
* movieName-שם הסרט
* removalDate-תאריך ירידת הסרט מהמסכים.
* Theater– יישות זאת אחראית על כל אולמות הקולנוע הנמצאים במסד הנתונים

ישות זאת הינה חזקה,.

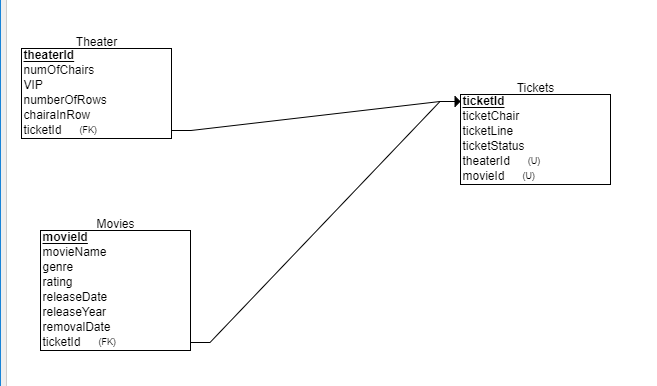
* theaterId – מספר מזהה של אולם הקולנוע (PK)
* numOfChairs- מספר הכסאות האולם הקולנוע
* VIP האם האולם הינו VIP או לא.
* numberOfRows- מספור שורות באולם הקולנוע
* chairaInRow-מספר כסאות בשורה באולם הקולנוע.

### קשרים

* TicketForTheater – הקשר בין Theater לבין Tickets. הקשר הינו חלש, כיוון שמקשר בין ישות חלשה לחזקה. הקשר הוא M:1 משום שיכול להיות לאולם אחד הרבה כרטיסים, אבל בקשת כרטיס אחד שייך למתקן אחד.
* TicketsForMovies – הקשר בין Theater לבין Movie. הקשר הינו חלש, כיוון שמקשר בין ישות חלשה לחזקה. הקשר הוא M:1 משום שיכול להיות לסרט אחד הרבה כרטיסים, אבל כרטיס אחד שייך לסרט אחד בלבד.

## תרשים DSD

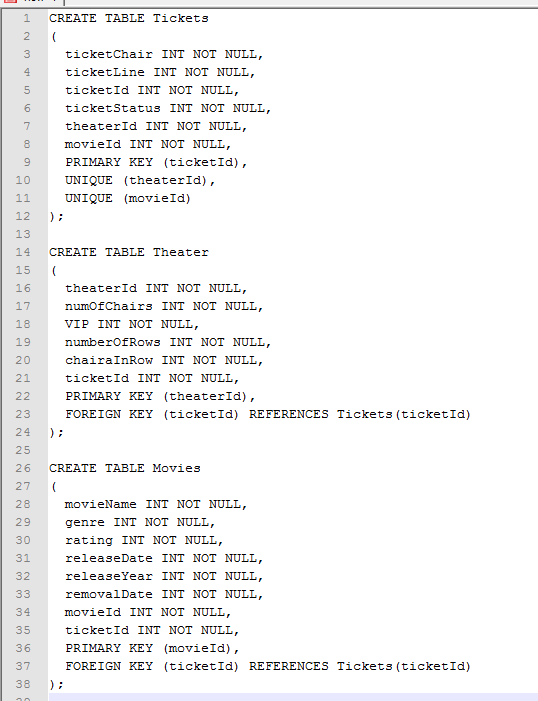
על פי תרשים ה- ERD ועל ידי הבנת הקשרים בין הישויות, יצרנו תרשים DSD עבור החלק שלנו במערכת: מחלקת מתקני המלון.



## יצירת הטבלאות

אחרי שהבנו כיצד בסיס הנתונים צריך להראות בצורה מדויקת, מה תכיל כל טבלה ומהם הקשרים בין כל הטבלאות, ניגשנו ליצירת הטבלאות בפועל בעזרת פקודות הcreate table.

יצרנו קוד לייצור הטבלאות באמצעות export SQL של האתר erdPlus, יצרנו קובץ SQL ואז העתקנו את קוד ה- SQL של כל טבלה אל תוכנת ה-plsql לשם יצירת הטבלאות בפועל(לשם הנוחות הראנו פה את הקוד כקובץ (JSON.



## הכנסת נתונים

על מנת לאכלס את הטבלאות שיצרנו בנתונים, השתמשנו ביבוא מידע מקובץ טקסט לתוך הטבלאות בבסיס הנתונים. יצרנו קובץ טקסט ובעזרת קוד בשפת C (הקוד מצורף ב- zip)אכלסנו אותו בנתונים.**בגיט של ארליך**

אחר כך השתמשנו בtext importer בתוכנת ה- PL/SQL כדי להכניס את הנתונים לטבלאות. כך לדוגמא, הכנסנו נתונים לתוך הטבלה supplyrequest:**ארליך עשה לטובת הכלל!!**

**שאילתות SQL**

### בחירה - SELECT

*עבור כל שאילתה צרפו צילום מסך של תוצאות ריצה וזמן ריצה. לעדכון- תראו לפני ואחרי. למחיקה- תראו צילום מסך שהתוכנה כתבה שנמחקו מספר שורות ולהוספה תראו שהתווספו השורות שהוספתם.*

*עבור כל שאילתה צרפו צילום מסך של תוצאות ריצה וזמן ריצה. לעדכון- תראו לפני ואחרי. למחיקה- תראו צילום מסך שהתוכנה כתבה שנמחקו מספר שורות ולהוספה תראו שהתווספו השורות שהוספתם.*

לאחר שיצרנו את בסיס הנתונים והכנסנו לתוכו מידע, כתבנו כמה שאילתות מעניינות על מנת לתשאל אותו.

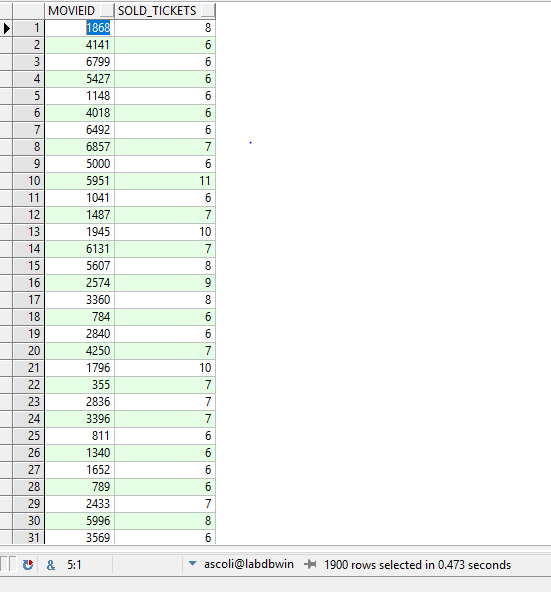
1. כתבנו שאילתה המחזירה טבלה של כל מזהי הסרטים שמספר כרטיסי הסרט שנקנו גדול מ-5 ואת מספר הסרטים עם מזהי סרט זה.

**select movieId, count(distinct(ticketId)) as sold\_tickets**

**from oehrlich.movies natural join oehrlich.ticketformovies natural join oehrlich.tickets**

**group by movieId**

**having count(ticketId) > 5**



1. כתבנו שאילתה המחזירה טבלה של כל מזהי כרטיסי הסרטים אשר מוצגים באולם מספר 2350 ושם הסרט הוא CyberThink

**select ticketId**

**from oehrlich.theater natural join oehrlich.ticketfortheatre natural join oehrlich.tickets natural join oehrlich.ticketformovies natural join oehrlich.movies**

**where theaterId = 2350 and movieName = 'CyberThink'**



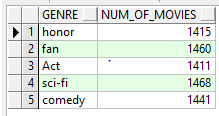
1. כתבנו שאילתה המחזירה את זאנר ומספר הסרטים אשר בקטגוריה הנ"ל אשר יצאו לאחר 1900 (ניתן לשנות את השנה)

**select genre, count(movieId) as num\_of\_movies**

**from oehrlich.movies**

**where releaseYear > 1900**

**group by genre**



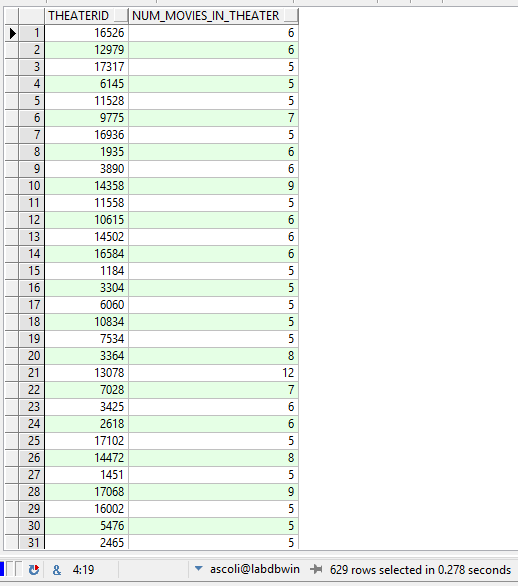
1. כתבנו שאילתה המחזירה טבלה של אולמות אשר מוצגים בהם יותר מ 5 סרטים שונים ואת מספר הסרטים באולם זה.

**select theaterId, count(distinct(movieId)) num\_movies\_in\_theater**

**from oehrlich.theater natural join oehrlich.ticketfortheatre natural join oehrlich.tickets natural join oehrlich.ticketformovies natural join oehrlich.movies**

**having count(distinct(movieId))>= 5**

**group by theaterId**

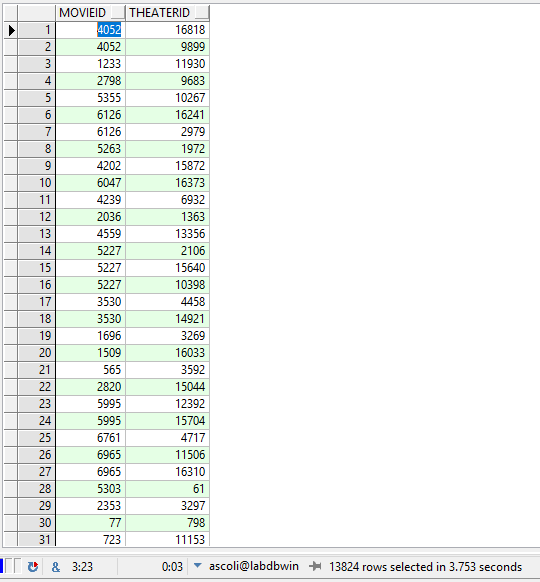


1. כתבנו שאילתה המחזירה את מזהי הסרטים ואת מזהי אולם הקולנוע אשר יש בו יותר מ-10 כסאות בשורה.

**select movieId, theaterId**

**from oehrlich.theater natural join oehrlich.ticketfortheatre natural join oehrlich.tickets natural join oehrlich.ticketformovies natural join oehrlich.movies**

**where chairaInRow > 10**

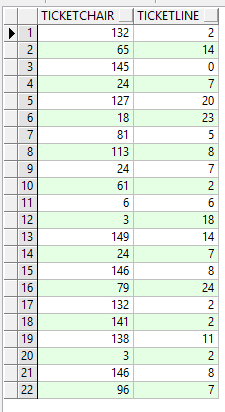


1. כתבנו שאילתה המחזירה את מספר הכסאות ואת המספר שבה הכסאות של מי שצופה בסרט NoeGen ולא באולם VIP.

**select ticketChair, ticketLine**

**from oehrlich.theater natural join oehrlich.ticketfortheatre natural join oehrlich.tickets natural join oehrlich.ticketformovies natural join oehrlich.movies**

**where movieName = 'Neogen' and VIP = 'NO'**

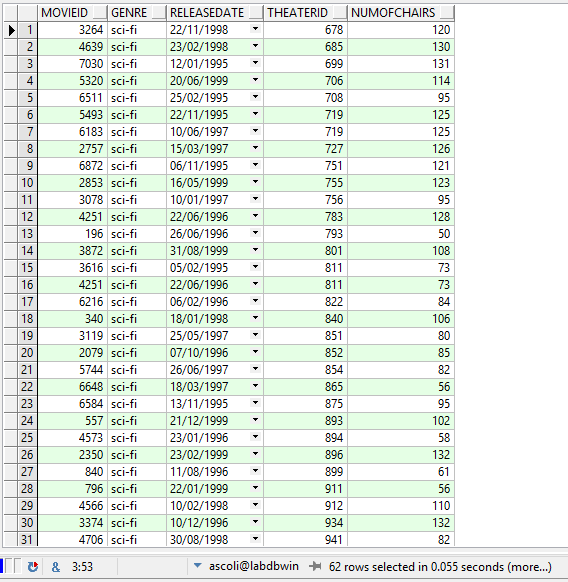


1. כתבנו שאילתה המחזירה את הטבלהה המכילה את פרטי הסרטים (מזהה סרט זאנר תאריך פרסום מספר אולם מספר כסאות באולם) שהג'אנר שלה היינו sci-fi ושיצא משנת 1995 ואלה.

**select movieId, genre, releaseDate, theaterId, numOfChairs**

**from oehrlich.theater natural join oehrlich.ticketfortheatre natural join oehrlich.tickets natural join oehrlich.ticketformovies natural join oehrlich.movies**

**where genre='sci-fi' and releaseDate > '01/Jan/1995'**



1. כתבנו שאילתה שמחזירה את מזהה הכרטיס, האולם ושם הסרט. כך ש גם הזאנר פעולה ומספר שורה הינו 4 וגם שהאולם מכיל יותר מ 200 כסאות.(אפשר לשנות את הערכים של משתני ה.(where

**select ticketId, theaterId, movieName**

**from oehrlich.theater natural join oehrlich.ticketfortheatre natural join (select ticketId, genre, movieName**

**from oehrlich.movies natural join oehrlich.ticketformovies natural join oehrlich.tickets**

**where genre = 'Act' and ticketLine = 4)**

**where numOfChairs> 200**

