**שאלות SQL**

1. להלן הגדרות יחסים בבסיס נתונים כלשהו:

**People(pid, name)**

**Chugim(cid, name, type)**

**InChug(pid, cid)**

הטבלה People מכילה מידע על אנשים (מספר מזהה ושם), הטבלה Chugim מכילה מידע על חוגים (מספר מזהה, שם וסוג) והטבלה InChug מכילה שורות המציינות את השתתפות האנשים בחוגים. ניתן להניח שהטבלאות אינן מכילות ערכי ,NULL שכל pid ב- InChug מופיע ב- People וכן שכל cid ב- InChug מופיע ב- Chugim. כתבו את השאילתות הבאות בSQL:

1. מצאו את שמות האנשים אשר **משתתפים בכל** החוגים מסוג "ספורט" **ואינם משתתפים באף** חוג מסוג "ריקוד".

select name from people P

where pid not in (select pid from chugim natural join inchug where type=’dance’)

and not exists ((select cid from chugim where type=’sport’)

minus

(select cid from chugim natural join inchug I

where I.pid = P.pid))

1. לכל סוג של חוג, מצאו את המספר המקסימאלי של משתתפים באחד החוגים מסוג זה וכן את המספר המינימאלי של משתתפים בחוג מסוג זה. **הבהרה:** כל שורה בתוצאה של השאילתה היא שלישיה. למשל, השורה (sport, 100, 3) מופיעה בתוצאה אם בחוג הגדול ביותר מסוג ספורט יש 100 אנשים ובחוג הקטן ביותר מסוג ספורט יש 3 משתתפים. לצורך שאלה זו ניתן להניח שלכל חוג בטבלה Chugim יש לפחות משתתף אחד.

select type, max(cc), min(cc)

from (select type, cid, count(pid) as cc

from chugim natural join inchug

group by type, cid) R

group by type

**Vote(pid, mid)  
Miflaga(mid, name, policy)**

הטבלה Vote מכילה מידע על אנשים (מספר מזהה), והמזהה של המפלגה עבורו הם הצביעו בבחירות לכנסת. הטבלה Miflaga מכילה מידע על מפלגות (מספר מזהה, שם ועמדה פוליטית policy). העמדה הפוליטית של מפלגה היא אחת מהמילים: שמאל, ימין, מרכז. ניתן להניח שהטבלאות אינן מכילות ערכי ,NULL ושכל mid ב-Vote מופיע ב-Miflaga.

1. לכל אחת משלושת העמדות הפוליטיות, מיצאו את שם המפלגה אשר השיגה את המספר הרב ביותר של קולות. (אם יש כמה מפלגות שלהן אותה עמדה ושהשיגו את המספר הרב ביותר של קולות, יש להחזיר את כולן.) שימו לב שהטבלה המוחזרת צריכה להכיל 2 עמודות: שם המפלגה והעמדה הפוליטית.

select name, policy

from Miflaga M1, Vote V1

where M1.mid = V1.mid

group by mid, name, policy

having count(pid) >= ALL(select count(pid)

from Vote V2, Miflaga M2

where V2.mid = M2.mid and M2.policy = M1.policy

group by V2.mid)

1. נניח שמספר המנדטים של מפלגה הוא מספר הקולות אותו השיגה המפלגה חלקי 30,000. (יתכן שמפלגה תקבל כמות לא שלמה של מנדטים). מצאו את כל הזוגות של מזהי מפלגות (m1,m2) כך ש-m1 ו-m2 יכולות להרכיב ביחד ממשלה, כלומר זוגות אשר מקיימים את 2 התנאים הבאים:
   1. או ש-m1 ו-m2 בעלות אותה עמדה פוליטית או שהעמדה הפוליטית של אחת מהן היא מרכז
   2. מספר המנדטים של m1 ו-m2 ביחד הוא גדול מ-60.

שימו לב שאם (m1,m2) מוחזר, אזי גם (m2,m1) יוחזר.

select M1.mid, M2.mid

from Miflaga M1, Miflaga M2

where M1.mid <> M2.mid and (M1.policy = M2.policy OR

M1.policy = ‘center’

OR M2.policy = ‘center’)

and 30,000 \* 60 < (select count(\*) from Votes V where V.mid = M1.mid or V.mid = M2.mid)

1. להלן הגדרות יחסים בבסיס נתונים כלשהו:

**Forecaster(forecasterId, name, rating)**

**Forecast(forecasterId, city, date, temp)**

**Weather(city, date, temp)**

הטבלה Forecaster מכילה מידע על חזאים, כולל מספר מזהה, שם ודירוג (מספר שלם בין 1 ל-5). הטבלה Forecast מכילה מידע על תחזיות של חזאים עבור ערים בתאריכים שונים. הטבלה Weather מכילה מידע על המזג אויר בפועל שנצפה בערים שונות ובתאריכים שונים. ניתן להניח שהטבלאות אינן מכילות ערכי ,NULL ושכל forecasterId ב-Forecast מופיע גם ב-Forecaster וכן שלכל חזאי בטבלת Forecaster יש לפחות תחזית אחת בטבלת Forecast.

1. מצאו את שמות כל החזאים אשר ממוצע ההפרש בין התחזיות שלהם לבין מזג האויר הנצפה הוא הקטן ביותר.

select name

from Forecaster where forecasterID in(

select forecasterId

from Forecast, Weather

where Forecast.city = Weather.city and Forecast.date = Weather.date

group by forecasterId

having avg(Forecast.temp-Weather.temp) <= ALL

(select avg(Forecast.temp-Weather.temp)

from Forecast, Weather

where Forecast.city = Weather.city and Forecast.date = Weather.date

group by forecasterId))

1. מצאו את המספר המזהה של כל אחד מהחזאים בדירוג 5 שנתן תחזית מזג אויר אחת לפחות לכל אחת מהערים עבורן יש תחזית כלשהי

select forecasterId

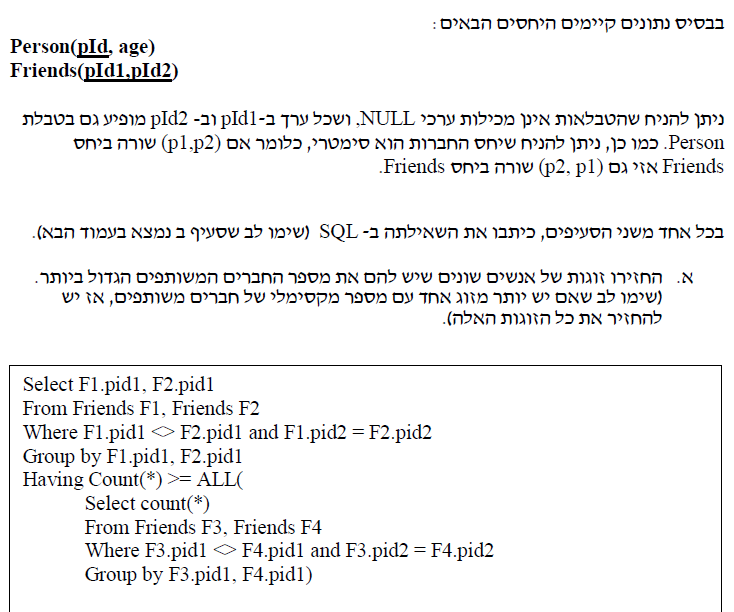
from Forecast F, Forecaster RF

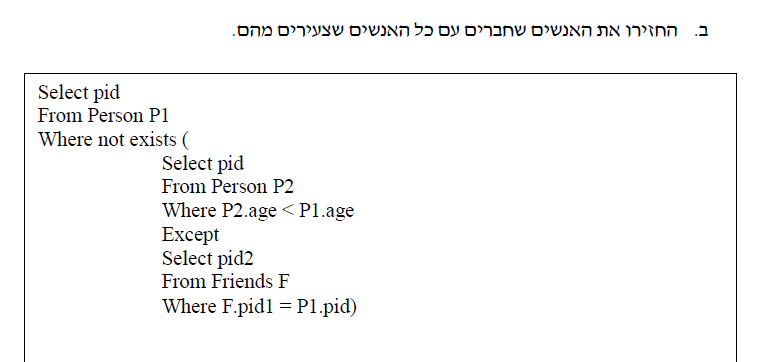
where F.forecasterId = RF.forecasterId and rating = 5 and not exists(

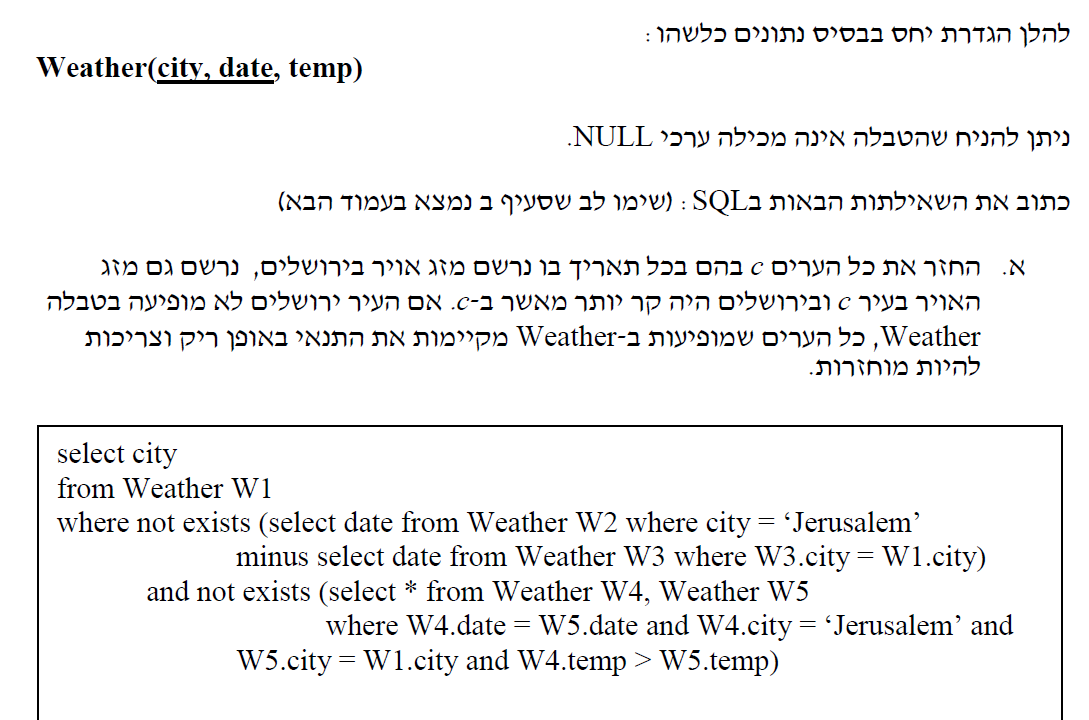
select city from Weather

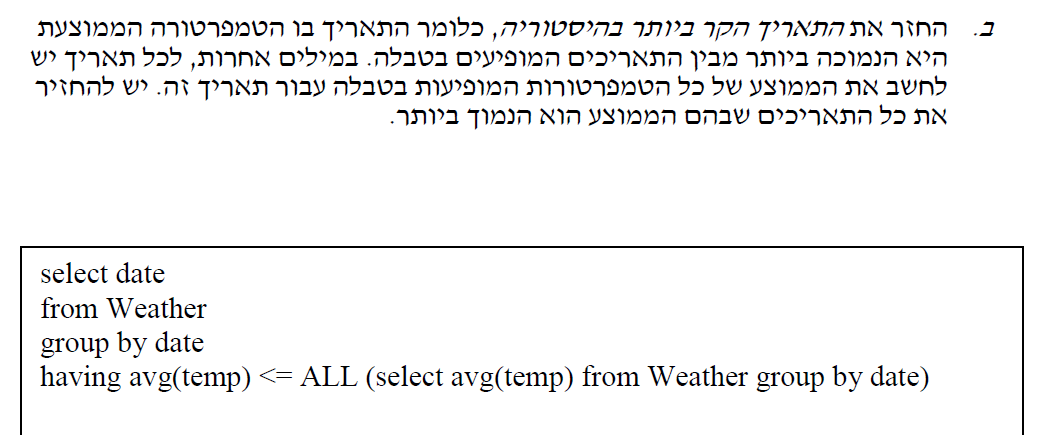
minus

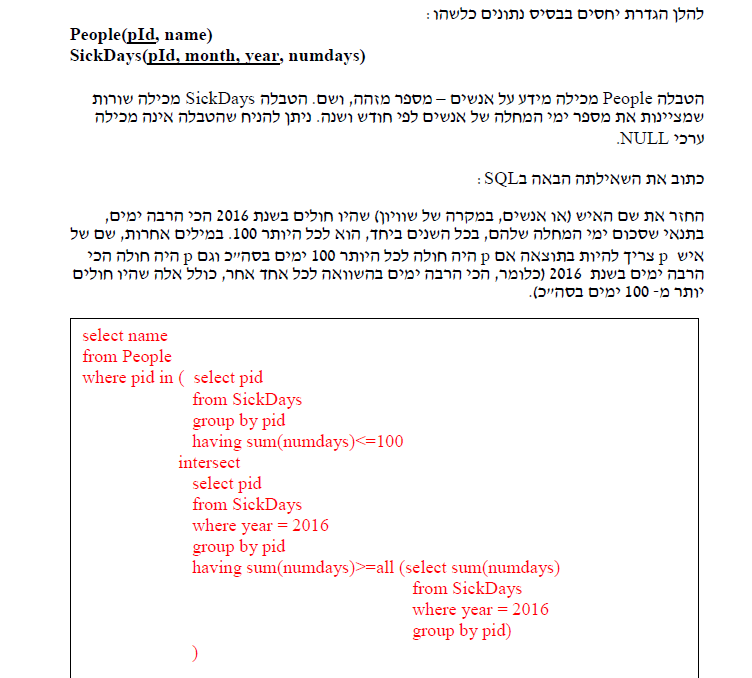
select city from Forecast F2 where F2.forecasterId = D.forecasterId)

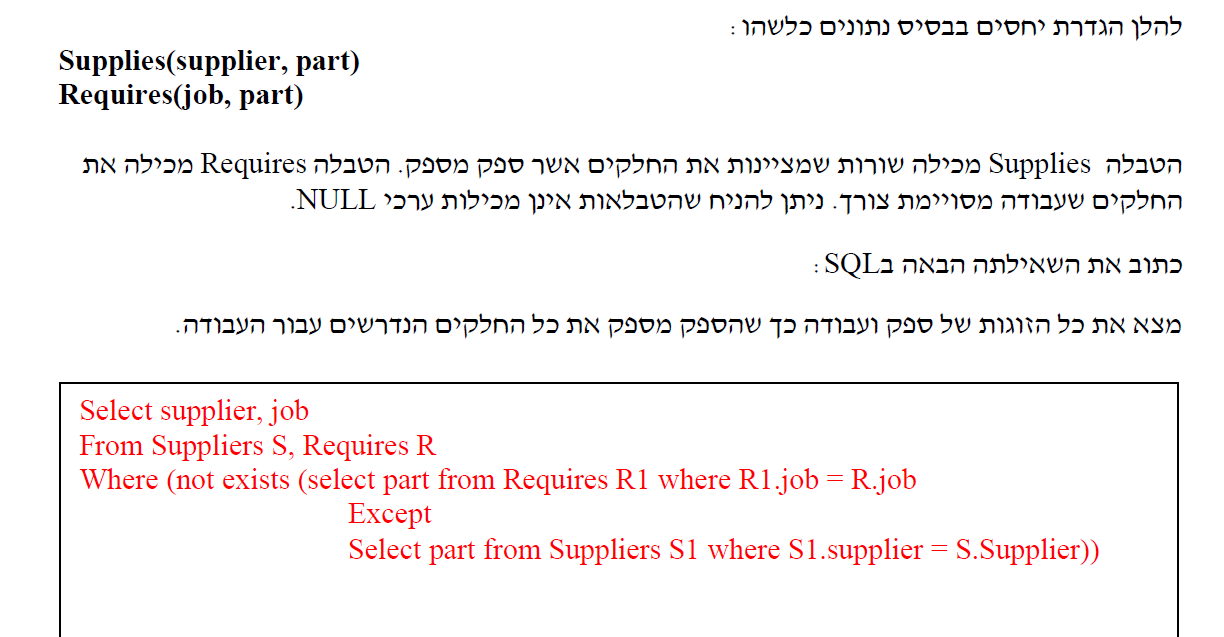


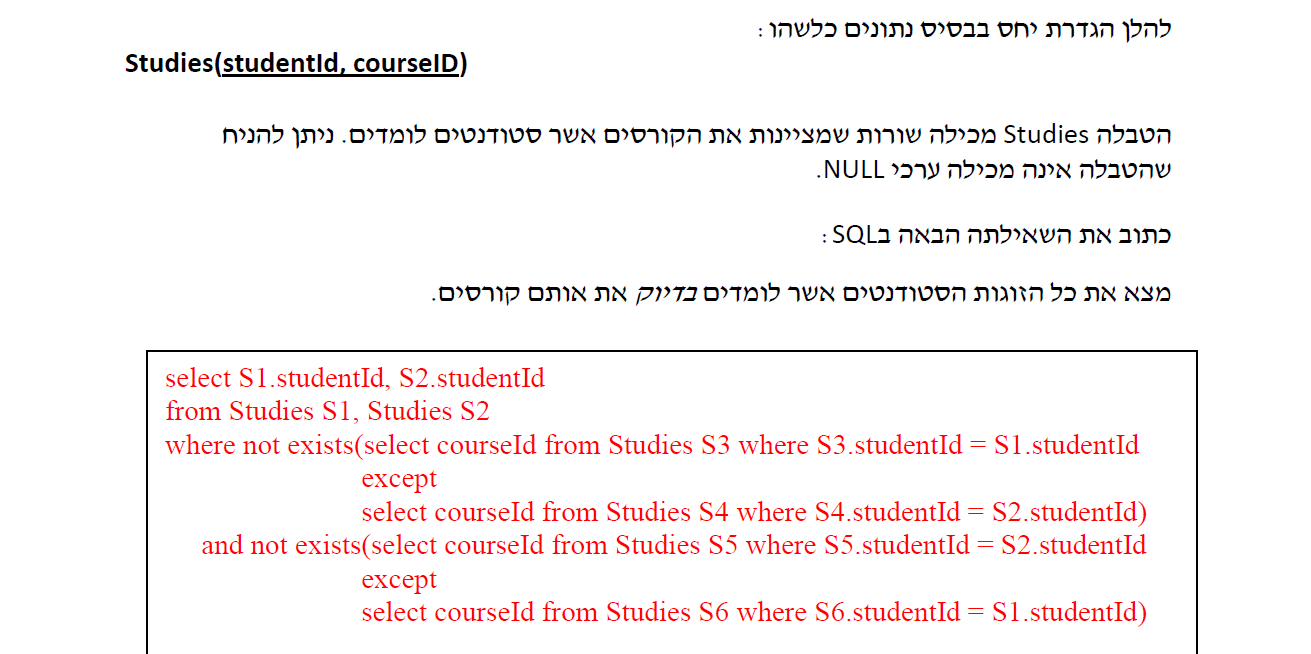


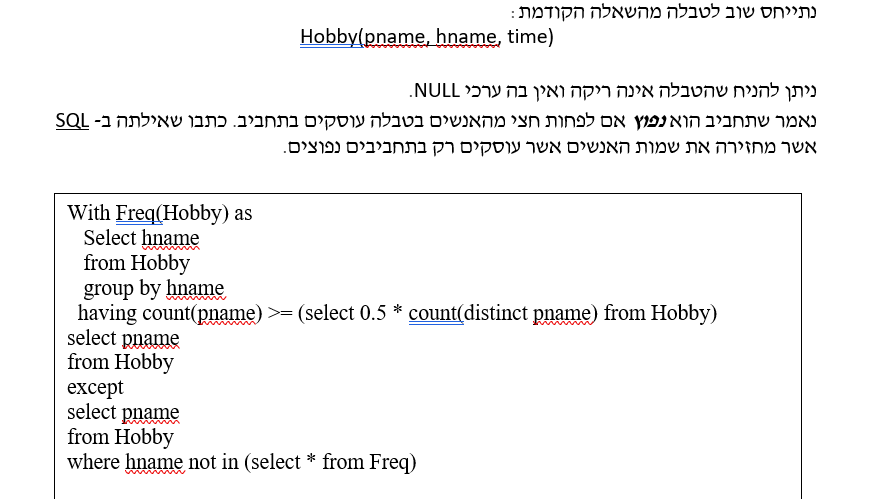


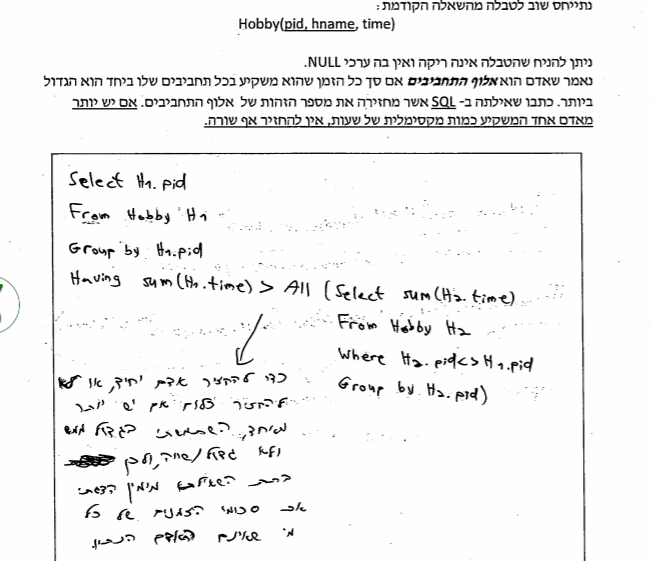


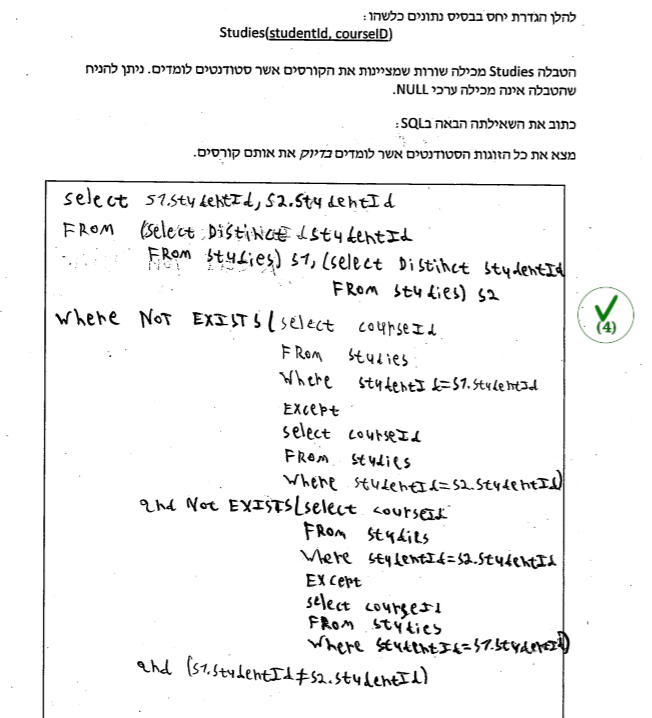


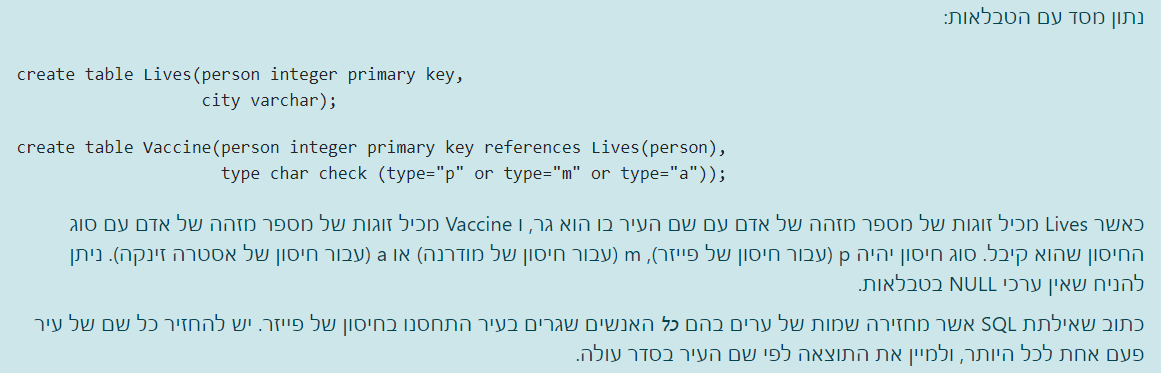
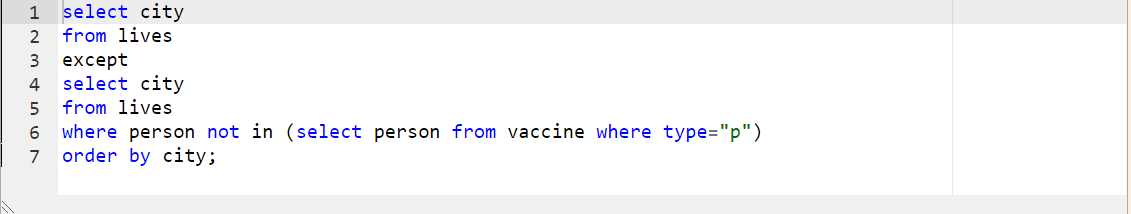


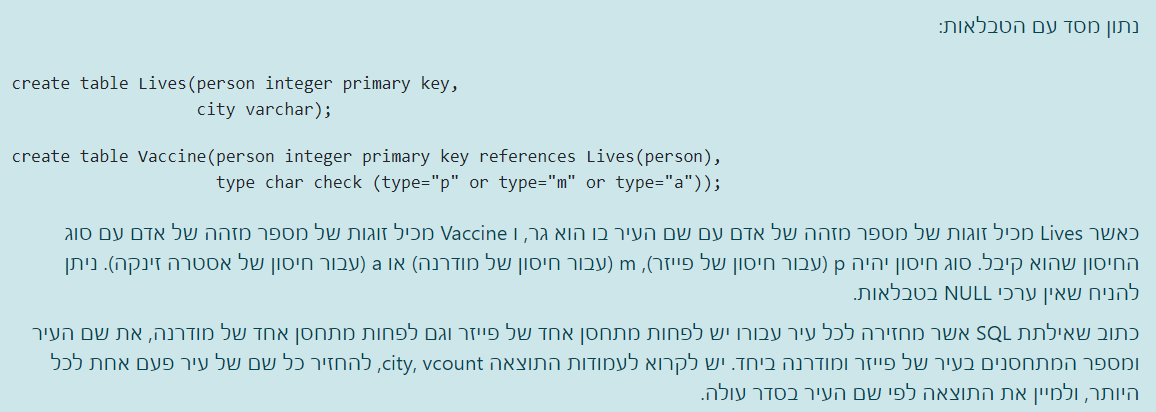












select P.city, P.c + M.c as vcount

from (select city, count(\*) as c from lives natural join vaccine where type="p" group by city having count(\*)>=1) P,

(select city, COUNT(\*) as c from lives natural join vaccine where type="m" group by city having count(\*)>=1) M

where P.city = M.city

order by P.city;