



מודל היחסים

- שליפת נתונים מתוך יחס אחד
- שליפת נתונים מתוך מספר יחסים ⇐
- מציאת ערך קיצוני
- פעולת החילוק
- תרגול

שליפת נתונים מתוך מספר יחסים



בכדי לשלוף נתונים מתוך מספר יחסים, יש שיטות שונות לחבר בין היחסים. נתחיל דווקא מהשיטות הפחות שימושיות מכיוון שהן הרבה יותר פשוטות להבנה:

1. איחוד יחסים ($\cup$) - מקביל לפעולת ה-union ב-sql (ערכים המופיעים או ביחס מימין או ביחס משמאל)
 2. חיתוך יחסים ($\cap$) - מקביל לפעולת ה-intersect ב-sql (ערכים המופיעים גם ביחס מימין וגם ביחס משמאל)
 3. הפרש יחסים ($-$) - מקביל לפעולת ה-except ב-sql (שימושי בעיקר למציאת ערכי קיצון - בהמשך) (כל הערכים המופיעים ביחס משמאל בתנאי שהם לא מופיעים ביחס מימין)
- * כל הפעולות מוגדרות רק כאשר היחסים מורכבים מאותה כמות של עמודות, עם אותו סוג מידע ומסודרות באותו סדר.

לדוגמה, השאילתה הבאה מחזירה את תעודות הזהות של כל הרופאים שגרים ברעננה ואת תעודות הזהות של כל הרופאים שגרים בכפר סבא:

$$\Pi_{did}(\sigma_{dcity = 'raanana'}(doctor)) \cup \Pi_{did}(\sigma_{dcity = 'kfar saba'}(doctor))$$

כמובן שהשאילתה שקולה לשאילתה הבאה:

$$\Pi_{did}(\sigma_{dcity = 'raanana' \vee dcity = 'kfar saba'}(doctor))$$

doctor (did, dname, dfamily, dphone, dcity, dstreet, dhouse, dspeciality, ddatestart, dsalary)
patient (pid, pname, pfamily, pbirthdate)
clinic (cid, ccity, caddress, cname)
treatment (tdate, did, pid, cid)

שליפת נתונים מתוך מספר יחסים



לדוגמה, השאילתה הבאה מחזירה את כל הפרטים של כל הרופאים שגרים ברעננה ומתמחים ברפואת ילדים:

$$(\sigma_{dcity = 'raanana'}(doctor)) \cap (\sigma_{dspeciality = 'children'}(doctor))$$

כמובן שהשאילתה שקולה לשאילתה הבאה:

$$\sigma_{dcity = 'raanana' \wedge dspeciality = 'children'}(doctor)$$

לדוגמה, השאילתה הבאה מחזירה את תעודות הזהות של כל הרופאים שלא ביצעו לעולם שום טיפול:

$$\Pi_{did}(doctor) - \Pi_{did}(treatment)$$

doctor (did, dname, dfamily, dphone, dcity, dstreet, dhouse, dspeciality, ddatestart, dsalary)
patient (pid, pname, pfamily, pbirthdate)
clinic (cid, ccity, caddress, cname)
treatment (tdate, did, pid, cid)

שליפת נתונים מתוך מספר יחסים



השיטות השימושיות יותר (והמורכבות יותר להבנה) לשליפת נתונים מתוך מספר יחסים הן:

1. מכפלה קרטזית ($\times$) - מקבילה לפעולה , ב-sql - מחברת כל שורה עם כל שורה (אם רוצים לגשת לתכונה כלשהי עם שם זהה בשני היחסים, חייבים לכתוב את שם היחס ממנו נלקחת התכונה ואז נקודה ואז את שם התכונה).
2. צירוף טבעי ($\bowtie$) - מקבילה לפעולת ה-natural join ב-sql - מחברת בין שורות שהערכים בהן זהים בתכונות בעלות שם זהה בשני היחסים.

מכפלה קרטזית מחברת כל שורה ביחס אחד עם כל שורה ביחס השני.
כל התכונות שיש ביחסים המחוברים יופיעו ביחס התוצאה ולכן אם יש תכונות שהשם שלהן זהה בשני היחסים, חייבים לכתוב מאיזה יחס אנחנו מעוניינים לקחת אותם, באמצעות כתיבת שם היחס ואז נקודה ואז שם התכונה, לדוגמה: doctor.did.
כמות השורות ביחס התוצאה של מכפלה קרטזית היא בדיוק מכפלה של כמות השורות ביחס האחד עם כמות השורות ביחס השני.

צירוף טבעי לעומת זאת מחבר אך ורק בין שורות שהערכים שנמצאים בתכונות בעלות אותו השם שווים זה לזה.
שימו לב: הצירוף מתייחס אך ורק לשם של התכונה! לא למשמעות שלה!

אם נצרף שני יחסים שבשניהם יש תכונה עם השם did, השורות שנקבל ביחס התוצאה הן רק השורות בהן הערכים של התכונה did זהים בשני היחסים.

במקרה זה, בניגוד למכפלה קרטזית, התכונה בעלת השם הזהה מופיעה רק פעם אחת ובכדי לשלוף תכונה מסוימת המופיעה בשני היחסים מיחס התוצאה, אין צורך לרשום מאיזה יחס התכונה הזו מגיעה.
אם יש מספר תכונות בעלות שם זהה, הצירוף יהיה רק לפי שורות שבהן כל הערכים של התכונות הללו זהים.

שליפת נתונים מתוך מספר יחסים



בואו נראה דוגמה לאיך נראית מכפלה קרטזית של שני יחסים:

tdate	clinicid	pid	did
17/1/2019	1	123	111
18/1/2019	2	456	222
22/1/2019	5	789	111
23/1/2019	7	543	222
25/1/2019	3	753	555
28/1/2019	2	456	222

×

ddatestart	dspeciality	dhouse	dstreet	dcity	dphone	dfamily	dname	did
1/8/1948	משפחה	1	האגס	ירושלים	02555555	לוי	יוסי	111
4/8/2011	נשים	5	בוגרשוב	תל אביב	03444444	כהן	משה	222
5/4/2005	נשים	4	רוטשילד	תל אביב	03448885	לוי	רבקה	555

=

treatment.				doctor.								
tdate	clinicid	pid	did	ddatestart	dspeciality	dhouse	dstreet	dcity	dphone	dfamily	dname	did
17/1/2019	1	123	111	1/8/1948	משפחה	1	האגס	ירושלים	02555555	לוי	יוסי	111
18/1/2019	2	456	222	1/8/1948	משפחה	1	האגס	ירושלים	02555555	לוי	יוסי	111
22/1/2019	5	789	111	1/8/1948	משפחה	1	האגס	ירושלים	02555555	לוי	יוסי	111
23/1/2019	7	543	222	1/8/1948	משפחה	1	האגס	ירושלים	02555555	לוי	יוסי	111
25/1/2019	3	753	555	1/8/1948	משפחה	1	האגס	ירושלים	02555555	לוי	יוסי	111
28/1/2019	2	456	222	1/8/1948	משפחה	1	האגס	ירושלים	02555555	לוי	יוסי	111
17/1/2019	1	123	111	4/8/2011	נשים	5	בוגרשוב	תל אביב	03444444	כהן	משה	222
18/1/2019	2	456	222	4/8/2011	נשים	5	בוגרשוב	תל אביב	03444444	כהן	משה	222
22/1/2019	5	789	111	4/8/2011	נשים	5	בוגרשוב	תל אביב	03444444	כהן	משה	222
23/1/2019	7	543	222	4/8/2011	נשים	5	בוגרשוב	תל אביב	03444444	כהן	משה	222
25/1/2019	3	753	555	4/8/2011	נשים	5	בוגרשוב	תל אביב	03444444	כהן	משה	222
28/1/2019	2	456	222	4/8/2011	נשים	5	בוגרשוב	תל אביב	03444444	כהן	משה	222
17/1/2019	1	123	111	5/4/2005	נשים	4	רוטשילד	תל אביב	03448885	לוי	רבקה	555
18/1/2019	2	456	222	5/4/2005	נשים	4	רוטשילד	תל אביב	03448885	לוי	רבקה	555
22/1/2019	5	789	111	5/4/2005	נשים	4	רוטשילד	תל אביב	03448885	לוי	רבקה	555
23/1/2019	7	543	222	5/4/2005	נשים	4	רוטשילד	תל אביב	03448885	לוי	רבקה	555
25/1/2019	3	753	555	5/4/2005	נשים	4	רוטשילד	תל אביב	03448885	לוי	רבקה	555
28/1/2019	2	456	222	5/4/2005	נשים	4	רוטשילד	תל אביב	03448885	לוי	רבקה	555



שליפת נתונים מתוך מספר יחסים

ואיך נראה צירוף טבעי:

tdate	clinicid	pid	did		ddatestart	dspeciality	dhouse	dstreet	dcity	dphone	dfamily	dname	did
17/1/2019	1	123	111	⋈	1/8/1948	משפחה	1	האגס	ירושלים	0255555	לוי	יוסי	111
18/1/2019	2	456	222		4/8/2011	נשים	5	בוגרשוב	תל אביב	0344444	כהן	משה	222
22/1/2019	5	789	111		5/4/2005	נשים	4	רוטשילד	תל אביב	0344885	לוי	רבקה	555
23/1/2019	7	543	222										
25/1/2019	3	753	555										
28/1/2019	2	456	222										

tdate	clinicid	pid	did	ddatestart	dspeciality	dhouse	dstreet	dcity	dphone	dfamily	dname
17/1/2019	1	123	111	1/8/1948	משפחה	1	האגס	ירושלים	0255555	לוי	יוסי
22/1/2019	5	789	111	1/8/1948	משפחה	1	האגס	ירושלים	0255555	לוי	יוסי
18/1/2019	2	456	222	4/8/2011	נשים	5	בוגרשוב	תל אביב	0344444	כהן	משה
23/1/2019	7	543	222	4/8/2011	נשים	5	בוגרשוב	תל אביב	0344444	כהן	משה
28/1/2019	2	456	222	4/8/2011	נשים	5	בוגרשוב	תל אביב	0344444	כהן	משה
25/1/2019	3	753	555	5/4/2005	נשים	4	רוטשילד	תל אביב	0344885	לוי	רבקה

אם אפשר, כמובן שנעדיף להשתמש בצירוף טבעי מאשר במכפלה קרטזית, גם כי זה חוסך מקום בזיכרון (לא רלוונטי לאלגברת יחסים אבל רלוונטי ל-sql), וגם כי זה חוסך כתיבה של תנאים מיותרים.

כמובן שאת פעולת הצירוף הטבעי תמיד נוכל לבצע באמצעות מכפלה קרטזית ותנאי או מספר תנאים.

שליפת נתונים מתוך מספר יחסים



לדוגמה, השאילתות הבאות זהות בתוצאה שלהן:

$$\Pi_{dname}(treatments \bowtie doctor) = \Pi_{dname}(\sigma_{doctor.did=treatment.did}(treatments \times doctor))$$

$$\Pi_{pname,pfamily}(treatments \bowtie patient) = \Pi_{pname,pfamily}(\sigma_{patient.pid=treatment.pid}(treatments \times patient))$$

$$\Pi_{pid}(\sigma_{tdate \geq 1.1.2022}(treatments \bowtie patient)) = \Pi_{patient.pid}(\sigma_{patient.pid=treatment.pid \wedge tdate \geq 1.1.2022}(treatments \times patient))$$

doctor (did, dname, dfamily, dphone, dcity, dstreet, dhouse, dspeciality, ddatestart, dsalary)
patient (pid, pname, pfamily, pbirthdate)
clinic (cid, ccity, caddress, cname)
treatment (tdate, did, pid, cid)

שליפת נתונים מתוך מספר יחסים



אם נרצה, אפשר לקרוא ליחסים שמהם אנחנו שולפים את הנתונים בשם חדש. זה נחמד כדי לקצר את השמות שלהם, אבל חובה להשתמש בזה כדי לבצע מכפלה קרטזית בין יחס לעצמו ולקרוא לכל אחד מהיחסים בשם שונה.

פעולה זה נקראת שכפול ($\rho_{new\ name(columns)}(R)$) והיא מקבילה לפעולה as ב-sql - לדוגמה, אם נרצה לקצר את השם של היחס "טיפולים":

$\rho_T(treatments)$

ואם נרצה לקרוא לתכונות של היחס ששכפלנו בשמות שונים: $\rho_{T(doc,patient,clinic,date)}(treatments)$

אם נרצה למצוא את כל המטופלים שהיו בשני טיפולים שונים אצל אותו רופא:

$\Pi_{T1.pid}(\sigma_{T1.pid=T2.pid \wedge T1.did=T2.did \wedge T1.tdate \neq T2.tdate}(\rho_{T1}(treatments) \times \rho_{T2}(treatments)))$

מצאו את כל המטופלים שהיו לפחות בשני טיפולים שונים במרפאות שונות:

doctor (did, dname, dfamily, dphone, dcity, dstreet, dhouse, dspeciality, ddatestart, dsalary)
patient (pid, pname, pfamily, pbirthdate)
clinic (cid, ccity, caddress, cname)
treatment (tdate, did, pid, cid)

שליפת נתונים מתוך מספר יחסים



אלגברת יחסים מאפשרת לנו ליצור יחס זמני. פעולה זו יכולה לפשט עבורנו בעיות מורכבות למספר בעיות קטנות, וכן יכולה לחסוך כתיבה מיותרת כאשר מעוניינים לחזור על אותם תנאים מספר פעמים.

פעולה זה נקראת השמה ($\leftarrow$) והיא מקבילה לפעולה with ב-sql - (תמיד מגדירים את היחס הזמני לפני שמנסים לקרוא לו).

לדוגמה, אם נרצה לשלוף את תעודות הזהות של כל רופאי המשפחה הגרים בתל אביב (כמובן שנוכל לעשות זאת באמצעות הגדרת מספר תנאים, אך בשביל הדוגמה נראה כאן דרך נוספת):

$$fd \leftarrow \sigma_{dspecialty="family"}(doctors)$$
$$\Pi_{did}(\sigma_{dcity="ta"}(fd))$$

השימוש בפעולה זו צריך להיעשות אך ורק אם הפעולה חוסכת בצורה משמעותית כתיבת קוד.