

תרגיל 2 : אלגברה רלציונית וSQL

תאריך הגשה : 23: 55, 22.11.20.

הוראות הגשה:

בתרגיל זה אתם נדרשים להגיש קובץ zip בודד שיכלול את הקבצים הבאים:

- ex2.pdf עם התשובות לשאלות בחלק של אלגברה רלציונית.
- q1.sql
- q2.sql
- q3.sql
- q4.sql
- q5.sql
- q6.sql
- README שמכיל שורה בודדת ובו ה-login של הסטודנט שמגיש את התרגיל. אם התרגיל מוגש בזוגות, על שורה זאת להכיל את שני ה-login מופרדים בפסיק.

שימו לב:

- נא לקרוא על הדרישות המנהליות של הקורס בלינק באתר הקורס כדי למלא את ההוראות להגשה של קבצים סרוקים!
- תרגיל מוקלד יזכה ב- 2 נקודות בונוס!

אלגברה רלציונית (50 נקודות):

שאלה א' (36 נקודות):

נתונים היחסים הבאים מתוך מסד נתונים של IMDb:

Movies (movieId, title, rating, year, duration, genre)

Actors (actorId, name, byear, dyear)

PlaysIn (movieId, actorId, character)

הערות:

- לכל סרט (Movie) יש מספר מזהה (movieId), כותרת (title), דירוג הסרט (rating), שנת יציאה (year), משך הסרט בדקות (duration) וז'אנר (genre).
- לכל שחקן (Actor) יש מספר מזהה (actorId), שם (name), שנת לידה (byear), ושנת פטירה (dyear). עבור שחקן חי, dyear הוא null.
- לכל משחק בסרט (PlaysIn) יש מספר מזהה של סרט ושחקן (movieId, actorId), ואת שם הדמות שהשחקן שיחק (character).

כתבו את השאילתות הבאות באלגברה רלציונית:

1. החזר את שמות כל השחקנים ששיחקו בסרט דמות ששמה "George".

$\pi_{name} \sigma_{character='George'}(Actors \bowtie PlaysIn)$

2. החזר את המספר המזהה וכותרת של כל הסרטים ששיחקו בהם שחקנים שמתו בשנת היציאה של הסרט, וגם הסרט הוא בז'אנר 'Documentary' או 'Drama'.

$$\pi_{movieid,title} \sigma_{year=dyear \wedge (genre='Documentary' \vee genre='Drama')} (Movies \bowtie PlaysIn \bowtie Actor)$$

3. החזר את המספר המזהה של כל השחקנים שגם השתתפו בסרט ששיחק בו השחקן 'Charles Chaplin', וגם השתתפו בסרט כלשהו באורך של יותר מ-90 דקות.

$$\pi_{actorid1} (\rho_{P1(movieid,actorid1,char1)} (PlaysIn) \bowtie PlaysIn \bowtie \sigma_{name='Charles Chaplin'} (Actors)) \\ \cap \pi_{actorid1} \sigma_{duration>90} (Movies \bowtie \rho_{P1(movieid,actorid1,char1)} PlaysIn)$$

4. החזר מספר מזהה של שחקנים ששיחקו בלפחות סרט אחד, וכל הסרטים בהם שחקו קבלו ערך rating גבוה מ-7.

$$\pi_{actorid} (PlaysIn) - \pi_{actorid} (\sigma_{rating \leq 7} (Movies \bowtie PlaysIn))$$

5. החזר את השמות של שחקנים ששיחקו בכל הסרטים ששיחק בהם השחקן עם id 100. (לא כולל השחקן עם id=100)

$$R1 = \pi_{actorid,movieid} (playsIn) \div \pi_{movieid} (\sigma_{actorid=100} (PlaysIn)) \\ result = \pi_{name} \sigma_{actorid \neq 100} (R1 \bowtie Actors)$$

6. החזר את שמות הסרטים ששיחקו בהם בדיוק 2 שחקנים מעל גיל 70 (היו בני יותר מ-70 כשיצא הסרט). שימו לב שניתן לכתוב בתנאי בחירה של אלגברה רלציונית ביטוי אלגברי, למשל, אם R(A,B) יחס, ניתן לכתוב ביטוי כגון $\sigma_{A>B} R$.

נמצא את כל הסרטים ששיחק בהם שחקן מעל גיל 70.

$$RA = \pi_{movieid,actorid} \sigma_{year-byear>70} (Movies \bowtie PlaysIn \bowtie Actors)$$

נגדיר:

$$\rho_{R1(movieid,actorid1)} RA \\ \rho_{R2(movieid,actorid2)} RA \\ \rho_{R3(movieid,actorid3)} RA$$

נמצא סרטים ששיחקו בהם לפחות 2 שחקנים

$$\rho_{RB(movieid,actorid1,actorid2)} \sigma_{actorid1 \neq actorid2} (R1 \bowtie R2)$$

נמצא סרטים ששיחקו בהם לפחות 3 שחקנים

$$\rho_{RC(movieid,actorid1,actorid2,actorid3)} (\sigma_{actorid1 \neq actorid3 \wedge actorid2 \neq actorid3} (\sigma_{actorid1 \neq actorid2} (R1 \bowtie R2) \\ \bowtie R3))$$

$$result = \pi_{title} (Movies \bowtie (\pi_{movieid} RB - \pi_{movieid} RC))$$

שאלה ב' (14 נקודות):

נתונים היחסים: R(A,B), S(B,C,D)
לכל אחד מהביטויים הבאים:

- אם הביטויים שקולים, הוכיחו זאת.
- אחרת, אם יש הכלה בכיוון אחד בלבד, הוכיחו את כיוון ההכלה והראו דוגמה נגדית לכיוון השני.

- אם אין הכלה באף כיוון, הראו דוגמא(ות) שמראות זאת.

$$1. \quad \pi_{A,D}(R \bowtie \pi_{B,D}\sigma_{C=8}(S)) \equiv? \pi_{A,D}\sigma_{C=8}(R \bowtie S)$$

$$\pi_{A,D}(R \bowtie S) \equiv \pi_A R \bowtie \pi_D S$$

הוכחה:

$$\begin{aligned} (a, d) \in \pi_{A,D}(R \bowtie \pi_{B,D}\sigma_{C=8}(S)) &\Leftrightarrow \\ \exists b (a, b, d) \in (R \bowtie \pi_{B,D}\sigma_{C=8}(S)) &\Leftrightarrow \\ \exists b ((a, b) \in R \text{ and } (b, d) \in \pi_{B,D}\sigma_{C=8}(S)) &\Leftrightarrow \\ \exists b ((a, b) \in R \text{ and } (b, 8, d) \in S) &\Leftrightarrow \\ \exists b (a, b, 8, d) \in R \bowtie S &\Leftrightarrow \\ \exists b (a, b, 8, d) \in \sigma_{C=8}(R \bowtie S) &\Leftrightarrow \\ (a, d) \in \pi_{A,D}\sigma_{C=8}(R \bowtie S) & \end{aligned}$$

$$2. \quad \pi_{A,D}(R \bowtie S) \equiv? \pi_A R \bowtie \pi_D S$$

$$\pi_{A,D}(R \bowtie S) \subseteq \pi_A R \bowtie \pi_D S$$

הוכחה:

$$\begin{aligned} (a, d) \in \pi_{A,D}(R \bowtie S) &\Rightarrow \\ \exists b, c (a, b, c, d) \in R \bowtie S &\Rightarrow \\ \exists b, c ((a, b) \in R \text{ and } (b, c, d) \in S) &\Rightarrow \\ (a) \in \pi_A R \text{ and } (d) \in \pi_D S &\Rightarrow \\ (a, d) \in \pi_A R \bowtie \pi_D S & \end{aligned}$$

דוגמא נגדית לכיוון שני:

$$R = \{(1,2)\}$$

$$S = \{(1,2,3)\}$$

$$\pi_{A,D}(R \bowtie S) = \emptyset$$

$$\pi_A R \bowtie \pi_D S = \{(1,3)\}$$

שאלות SQL (50 נקודות):

בכל התשובות לשאלות בחלק זה:

- השתמשו ב-SELECT DISTINCT כדי למנוע כפילויות בתשובות (אם כפילויות עלולות להווצר בתשובה).
- **שימו לב:** בכל סעיף כתוב באיזה סדר למיין את התוצאות וכן את שמות העמודות בתוצאה.

באתר הקורס יש קובץ create.sql המכיל הגדרות עבור הטבלאות וקובץ drop.sql המכיל פקודות המוחקות את הטבלאות. כמו כן, נתונים הקבצים:

actors.csv -
movies.csv -
playsIn.csv -

הקבצים מכילים מידע חלקי אך אמיתי אודות סרטים ושחקנים מהאתר IMDb. את המידע המלא ניתן למצוא ב-<https://www.imdb.com/interfaces>.
הקבצים שלנו מכילים מידע על כ-10,000 סרטים, שעבר "ניקוי" והתאמה לצורך התרגיל.

ניתן למצוא את הקבצים גם במערכת המחשבים במעבדה בתיקה:

~ db/data/

ניתן להעתיק אותם לתיקיה שלכם.

על מנת לבדוק את התרגיל שלכם, יש ליצור את הטבלאות בעזרת create.sql, ולטעון לתוכן נתונים בעזרת הפקודות

```
cat movies.csv | psql -h dbcourse public -c "copy movies from STDIN DELIMITER ',' CSV HEADER"
```

```
cat actors.csv | psql -h dbcourse public -c "copy actors from STDIN DELIMITER ',' CSV HEADER"
```

```
cat playsIn.csv | psql -h dbcourse public -c "copy playsIn from STDIN DELIMITER ',' CSV HEADER"
```

כתבו את השאילתות הבאות ב-SQL. שם הקובץ שבו צריכה להופיע התשובה לכל שאלה נמצא בתחילת השאלה. שימו לב ששאלות 1-6 דומות לשאלות בחלק של אלגברה, אך ישנם שינויים שיש לשים לב אליהם בשאלות 1,5,6.

1. (q1.sql) מצא את שמות כל השחקנים ששיחקו בסרט דמות ששמה **מתחיל ב- "George"**, **והחזר את שמות השחקנים ושם הדמות ששיחקו.**

יש להחזיר טבלה עם העמודות name, character, ממויין מיון ראשוני לפי actorName ומיון שניוני לפי character.

```
select distinct name, character
from PlaysIn natural join actors
where character like 'George%'
order by name, character;
```

2. (q2.sql) החזר את המספר המזהה וכותרת של כל הסרטים ששיחקו בהם שחקנים שמתו בשנת היציאה של הסרט, וגם הסרט הוא בז'אנר 'Documentary' או 'Drama'. יש להחזיר טבלה עם העמודות movieid, title, ממויין מיון ראשוני לפי movieid ומיון שניוני לפי title.

```
select distinct movieid, title
from actors A natural join playsIn P natural join movies M
where A.dyear=M.year and (genre='Documentary' or genre='Drama')
order by movieid, title;
```

3. (q3.sql) החזר את המספר המזהה של כל השחקנים שהשתתפו בסרט ששיחק בו השחקן 'Charles Chaplin', וגם השתתפו בסרט כלשהו באורך של יותר מ-90 דקות. יש להחזיר טבלה עם העמודות actorId, ממויין לפי actorId.

```
select P1.actorid
from playsIn P1, playsIn P2 natural join actors
where P1.movieid=P2.movieid and name='Charles Chaplin'
```

intersect

```
select actorid
from playsIn natural join movies
where duration>90
```

order by P1.actorid;

4. **(q4.sql)** החזר מספר מזהה של שחקנים ששיחקו רק בסרטים עם rating גבוה מ-7, ושיחקו לפחות בסרט אחד. יש להחזיר טבלה עם העמודה actorId, ממויין לפי actorId.

```
select actorid
from playsin
```

except

```
select actorid
from playsIn natural join movies
where rating <=7
```

order by actorid;

5. **(q5.sql)** החזר זוגות של actorid1, actorid2 כך ש :

- actorid2 משחק לפחות בסרט אחד
- השחקן עם מספר מזהה actorid1 משחק (לפחות) בכל הסרטים שמשחק בהם השחקן עם מספר מזהה actorid2, actorid2 ≠ actorid1
-

יש להחזיר טבלה עם העמודות הנקראות actorid1, actorid2 ממויין לפי actorid1 ומיון שניוני לפי actorid2.

```
select distinct p1.actorid as actorid1, p2.actorid as actorid2,
from playsIn p1, playsIn p2
where p1.movieid=p2.movieid and p1.actorid <>p2.actorid and not exists
```

```
(select movieid
from playsIn p3
where p3.actorid=p2.actorid and movieid not in
(select movieid
from playsin p4 where actorid=p1.actorid))
```

order by p1.actorid, p2.actorid;

6. **(q6.sql)** החזר מספר מזהה וכותרת של כל הסרטים שבהם יש לפחות שני שחקנים שהפרש גילם הוא יותר מ-60. יש להחזיר טבלה עם העמודות title, movieid, ממויין מיון ראשוני לפי movieid ומיון שניוני לפי title.

```
select distinct M.movieid, M.title
from Playsin P1 natural join actors A1, playsin P2 natural join actors A2 , Movies M
where P1.movieid=P2.movieid and M.movieid=P1.movieid and A1.byear-A2.byear>60
order by movieid, title;
```