תרגיל 3 : SQL מתקדם ואינדקסים

תאריך הגשה: 23:55, 18.05.25.

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

- ex3.pdf עם התשובות לשאלות בחלק א שאלה 2 סעיף א ובחלק ב לכל השאלות.
- q1.sql
- q2.sql
- q3.sql
- q4.sql
- q5.sql
- q6.sql
- example.sql – עבור חלק א שאלה 2 סעיף ב.
- correct.sql – עבור חלק א שאלה 2 סעיף ג.
- 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.

תזכורת: יש להגיש תרגיל מוקלד בלבד.

חלק א: שאילתות **SQL** (50 נקודות)

שאלה 1:

נתונים היחסים הבאים מתוך מסד הנתונים של הכנסת (זהים ליחסים מתרגיל 2):

members (uid, name, occupation, birthPlace, gender, educatedAt, language, birthYear)

knessets (number, startYear, endYear)

memberInKnesset (number, uid, party)

הערות:

- בטבלה של חברי כנסת (members) יש את המידע על מי שהיו חברית כנסת מהכנסת הראשונה ועד כנסת מספר 24:
  - uid – מספר מזהה יחודי של חבר הכנסת במסד הנתונים.
  - name – שם חבר הכנסת.
  - occupation – המקצוע של חבר הכנסת.
  - birthPlace – עיר הלידה של חבר הכנסת.
  - gender – המגדר של חבר הכנסת.
  - educatedAt – המוסד האקדמי בו למד חבר הכנסת.
  - language – השפה העיקרית בה מדבר חבר הכנסת.
  - birthYear – שנת הלידה של חבר הכנסת.

- בטבלה של הכנסות (knessets) יש את המידע לגבי הכנסות מספר 1 עד 24:
  - number – מספר הכנסת.

- startYear – שנת התחלת הכהונה.
- endYear – שנת סיום הכהונה.
-

- בטבלה של חברים בכנסת (memberInKnesset) יש את המידע על איזה חבר כיהן באיזו כנסת:
  - number – מספר הכנסת.
  - uid – מספר מזהה של חבר הכנסת.
  - party – המפלגה אותה ייצג החבר בכנסת.
  שימו לב שהמפלגה היא גם חלק מהמפתח של היחס מכיוון שישנם חברי כנסת שעברו בין מפלגות באותה כנסת.

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

- members.csv
- knessets.csv
- memberInKnesset.csv

הקבצים מבוססים על מידע שנילקח מאתר kaggle בכתובת https://www.kaggle.com/datasets/guybarash/israeli-parliament-knesset-members. בקבצים שמספוקים לכם מופיעות רק חלק מהעמודות שהופיעו במידע המקורי באתר כדי לפשט את התרגיל.

ניתן למצוא את הקבצים גם במערכת המחשבים במעבדה בתיקיה:

~db/data/ex2/

ניתן להעתיק אותם לתיקיה שלכם.

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

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

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

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

בכל התשובות לשאלות בחלק זה:

- השתמשו ב SELECT DISTINCT כדי למנוע כפילויות בתשובות (אם כפילויות עלולות להיווצר בתשובה).
- שימו לב: בכל סעיף כתוב באיזה סדר למיין את התוצאות וכן את שמות העמודות בתוצאה.
- אין סיבה להשתמש ב**view** כדי לענות על השאלות, אפשר להתשמש בטבלה זמנית עם **with** במקרה הצורך.
- אין צורך לעגל נתונים שיצאו עם נקודה עשרונית.

כתבו את השאילתות הבאות בSQL. שם הקובץ שבו צריכה להופיע התשובה לכל שאלה נמצא בתחילת השאלה.

1. **(q1.sql)** לכל כנסת החזר את מספר המפלגות השונות בהן כיהנו חברי כנסת באותה הכנסת. יש להחזיר טבלה עם העמודות (number, partyCount) ממוינת לפי number.

2. **(q2.sql)** לכל כנסת החזר את הגיל הממוצע (בתחילת הכהונה) של חברי הכנסת שכיהנו באותה הכנסת. יש להחזיר טבלה עם העמודות (number, avgAge) ממוינת לפי number.

3. **(q3.sql)** נאמר שחבר כנסת הוא מתמיד נאמן, אם כיהן לפחות בחמש כנסות שונות (לאו דווקא ברצף) ובנוסף, כיהן עבור מפלגה אחת בלבד לאורך כל כהונותיו בכנסת. החזר את שמות כל חברי הכנסת המתמידים-נאמנים. יש להחזיר טבלה עם העמודה (name) ממוינת לפי name.

4. **(q4.sql)** לכל כנסת, החזר את המפלגה הכי גדולה (כלומר שכיהנו מטעמה בכנסת זו הכי הרבה חברי כנסת) ואת מספר חברי הכנסת מאותה מפלגה באותה הכנסת. יש להחזיר טבלה עם העמודות (number, party, memberCount) ממוינת לפי number ומיון שניוני לפי party. אם יש כמה מפלגות גדולות ביותר באותה כנסת, יש להחזיר את כולם, כל אחת בשורה נפרדת.

5. **(q5.sql)** נאמר שמפלגה היא מקדמת נשים אם יש כנסת בה לפחות 30% מחברי הכנסת באותה מפלגה היו נשים. מצאו את כל המפלגות שהיו מקדמות נשים בלפחות כנסת אחת. יש להחזיר טבלה עם העמודות (party, number, femalePercent) ממוינת לפי party ומיון שניוני לפי number.
<u>הערה</u>: femalePercent צריך להיות מספר האחוזים ולא שבר עשרוני. כלומר אם יש 30% נשים, הערך יהיה 30 ולא 0.3.

6. **(q6.sql)**. נאמר שהמרחק בין שני חברי כנסת m1 ו-m2 הוא 1 אם שניהם כיהנו באותה הכנסת עבור אותה המפלגה. בהמשך לכך, אם חבר כנסת m3 כיהן באותה הכנסת עבור אותה המפלגה כמו m2 (אך לא עם m1), נוכל לומר שחברי הכנסת m1 ו m3 הם במרחק 2 .למשל, גאולה כהן כיהנה בכנסת ה-8 יחד עם מנחם בגין במפלגת הליכוד, ולכן היא במרחק 1 ממנו. חנן פורת לא כיהן בכנסת ה-8, אבל כן כיהן יחד גם גאולה כהן בכנסת ה-10 במפלגת תחיה לכן הוא במרחק 2 ממנחם בגין.
כתבו שאילתה <u>רקורסיבית</u> אשר מחזירה את כל חברי הכנסת שהמרחק שלהם ממנחם בגין (Menachem Begin) <u>גדול מ- 3</u>.
יש להחזיר טבלה עם שתי עמודות (uid, name) ממויינת לפי uid
<u>הערה:</u> אם בין שני חברי כנסת לא מוגדר מרחק (אם אי אפשר להגיע מאדם אחד לאדם אחר) - זה נחשב למרחק אינסוף - ולכן גדול מ-3.

<u>שאלה 2:</u>

כפי שרובכם בוודאי כבר יודעים, ChatGPT הוא צ'אטבוט שפותח על-ידי חברת OpenAI על בסיס מודל השפה שלה. בין היכולות הרבות שלו, אפשר להשתמש ב-ChatGPT כדי לדמות כתיבת קוד בשפות שונות, ואף כדי לדמות כתיבת שאילתות ב-SQL.
אבל כמו שמחשבון הוא מאוד שימושי, אך לא פותר אותנו מלדעת מתמטיקה, כך גם ChatGPT יכול להיות כלי מאוד שימושי, אך אינו יכול להחליף את הצורך של מתכנתים לדעת לכתוב ולהבין קוד בעצמם (או את הצורך של סטודנטים לעבור את המבחן).

בשאלה זו נדגים את הבעייתיות בשימוש בChatGPT לכתיבת שאילתת SQL בלי לדעת לכתוב שאילתה נכונה בעצמנו.

באחד ממועדי הבחינה של הקורס, מופיעה השאלה הבאה:

נתון מסד נתונים עם טבלה עם מידע על תרומות:

```
create table donors(name varchar, cause varchar, amount integer);
```

שמכילה שלשות:

- name: שם התורם
- cause: העמותה שקבלה את הכסף
- amount: סכום התרומה

ניתן להניח שאין שני תורמים שונים עם אותו שם, וכן אין שני עמותות שונות עם אותו שם.
נאמר שתורם n1 *זהה במטרותיו* לתורם *n2* אם שניהם תרמו לאותן עמותות בדיוק (לא משנה מספר התרומות לכל עמותה).

יש לכתוב שאילתת SQL אשר מחזירה את כל הזוגות של שמות תורמים (n1, n2) כך ש-n1 זהה במטרותיו ל-n2, וגם השם של n1 מופיע לפני השם של n2 לפי סדר אלפביתי. יש להחזיר את התוצאה, ללא כפילויות, ממוינת לפי שם התורם n1 ואחר כך לפי n2, בסדר עולה.

כאשר השאלה תורגמה לאנגלית ונשלחה ל ChatGPT, התקבלה התשובה הבאה:

```
SELECT d1.name AS n1 ,d2.name AS n2

FROM donors d1, donors d2

WHERE d1.cause = d2.cause and d1.name < d2.name

GROUP BY n1, n2

HAVING COUNT(DISTINCT d1.cause) = COUNT(DISTINCT d2.cause)

ORDER BY n1, n2
```

התשובה שגויה.

א. הסבר במילים שלך מה השאילתה של ChatGPT מחשבת.

ב. תן דוגמה מינימאלית של טבלה שמדגימה שהשאילתה מחזירה תשובה שגויה.
את התשובה יש להגיש בצורה של פקודה או פקודות insert לטבלה donors בקובץ בשם example.sql.
תוכלו להשתמש בקישור הבא ל fiddle כדי לנסות דוגמאות שונות ולהריץ עליהן את השאילתה השגויה:
http://sqlfiddle.com/#!5/e27122/2

ג. כתוב שאילתה נכונה עבור השאלה. את הקוד יש להגיש בקובץ correct.sql

חלק ב: אינדקסים (50 נקודות): להגשה בכתב בקובץ (**ex3.pdf**)

בחלק זה אנחנו נשתמש בטבלה members מחלק א שאלה 1 המובאת פה שוב לייתר נוחות :

members (uid, name, occupation, birthPlace, gender, educatedAt, language, birthYear)

בסעיפים הבאים, יש לכתוב הסבר לדרך הפתרון, ולהדגיש את התוצאה הסופית של כל חישוב!

בכל סעיף אפשר להשתמש רק באינדקס הנתון באותו סעיף ולא באינדקסים שהוגדרו בסעיפים קודמים.

הנחות:

- גודל בלוק הוא 1,024 בייטים.
- בטבלה members יש 1,000 שורות,
- כל שורה תופסת 64 בייטים.
- התכונה birthYear תופסת 4 בייט.
- התכונה uid תופסת 4 בייט.
- התכונה language תופסת 10 בייט.
- מצביע תופס 4 בייט.
- הערכים ב birthYear בטבלה members מתפלגים אחיד בטווח [1900,2000)
- הערכים ב language בטבלה מחולקים ל5 קטגוריות באופן אחיד.

1. נתונה השאילתה הבאה:

```
SELECT DISTINCT “exists”
FROM members
WHERE birthYear > 1990
```

א. מה עלות חישוב השאילתה בהנחה שאין אינדקסים על הטבלה?

כעת, נתון האינדקס הבא על הטבלה:

```
CREATE index on members(birthYear)
```

ב. מה תהיה דרגת הפיצול האופטימלית של האינדקס?

ג. מה תהיה עלות חישוב השאילתה באמצעות האינדקס, בהנחה שדרגת הפיצול היא זו שחושבה בסעיף הקודם?

2. נתונה השאילתה הבאה:

```
SELECT avg(birthYear)
FROM members
WHERE birthYear > 1990
```

ונתון אותו האינדקס כמו בשאלה 1:

```
CREATE index on members(birthYear)
```

מה תהיה עלות חישוב השאילתה באמצעות האינדקס, בהנחה שדרגת הפיצול היא זו שחושבה בסעיף הקודם?

3. נתונה השאילתה הבאה:

```
SELECT name
FROM members
WHERE uid=58311
```

ונתון האינדקס הבא על הטבלה:

```
CREATE index on members(uid)
```

א. מה תהיה דרגת הפיצול האופטימלית של האינדקס?

ב. מה תהיה עלות חישוב השאילתה באמצעות האינדקס, בהנחה שדרגת הפיצול היא זו שחושבה בסעיף הקודם?

4. נתונה השאילתה הבאה:

```
SELECT avg(birthYear)
FROM members
WHERE language = 'Hebrew'
```

ונתון האינדקס הבא על הטבלה :

```
create index on members(language)
```

א. מה תהיה דרגת הפיצול האופטימלית של האינדקס?

ב. מה תהיה עלות חישוב השאילתה באמצעות האינדקס, בהנחה שדרגת הפיצול היא זו שחושבה בסעיף הקודם?

כעת, נתון האינדקס הבא על הטבלה:

```
create index on members(language,birthYear)
```

ג. מה תהיה דרגת הפיצול האופטימלית של האינדקס?

ד. מה תהיה עלות חישוב השאילתה באמצעות האינדקס, בהנחה שדרגת הפיצול היא זו שחושבה בסעיף הקודם?

בהצלחה!