תרגיל 3 : SQL מתקדם ואינדקסים

תאריך הגשה : 01.05.22 ,23:55.

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

- ex3.pdf עם התשובות לשאלות בחלק ב : אינדקסים.
- 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 אחר ההוראות להגשה של קבצים סרוקים!
- תרגיל מוקלד יזכה ב- 2 נקודות בונוס!

נתונים היחסים הבאים מתוך מסד נתונים של האתר CSRankings (זהים ליחסים בתרגיל 2) :

authors (name, conference, year, institution, count, adjustedcount)

conferences (conference, area, subarea)

institutions (institution, region, country)

הערות :

- בטבלה של מחברים (authors) יש את המידע על פרסומים של מחברים בכנסים שונים :
  - name – שם המחבר.
  - conference – שם הכנס שבו הוא פרסם.
  - year – השנה שבה פורסם המאמר בכנס.
  - institution – שם המוסד האקדמי של המחבר.
  - count – מספר המאמרים שהמחבר פרסם באותו הכנס.
  - adjustedcount – מספר הפרסומים היחסי של המחבר בכנס. למשל אם פרסם מאמר אחד והיה אחד משני כותבים הספירה היחסית תהיה 0.5. אם היו שלשה כותבים למאמר ...0.33 וכו'.
- בטבלה של הכנסים (conferences) יש את המידע לגבי הכנסים :
  - conference – שם הכנס
  - area – תחום המחקר של הפרסומים בכנס
  - Subarea – תת-תחום המחקר.
- בטבלה של מוסדות (institutions) יש מידע על מוסדות אקדמים :
  - institution – שם המוסד
  - region – אזור גיאוגרפי בעולם.
  - country – המדינה בה נמצא המוסד מיוצגת בתקציר ע"י שני אותיות. למשל ישראל היא il.

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

- generated-author-info.csv
- conferences.csv
- country-info.csv

הקבצים מכילים מידע על מחברים, פרסומים, כנסים ומוסדות אקדמים. המידע משמש את האתר http://csrankings.org/ לדירוג מוסדות אקדמים בתחום מדעי המחשב. כל הקבצים זהים לאלו שסופקו בתרגיל 2.

את המידע המלא ניתן למצוא בלינקים הבאים:

https://raw.githubusercontent.com/cohensara/csrankings/main/conferences.csv
https://raw.githubusercontent.com/emeryberger/CSrankings/gh-pages/generated-author-info.csv
https://raw.githubusercontent.com/emeryberger/CSrankings/gh-pages/country-info.csv

ניתן למצוא את הקבצים גם במערכת המחשבים במעבדה בתיקיה (הקבצים זהים לאלו של תרגיל 2):

~ 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 generated-author-info.csv | psql -h dbcourse public -c "copy authors from STDIN DELIMITER ',' CSV HEADER"
cat conferences.csv | psql -h dbcourse public -c "copy conferences from STDIN DELIMITER ',' CSV HEADER"
cat country-info.csv | psql -h dbcourse public -c "copy institutions from STDIN DELIMITER ',' CSV HEADER"
```

חלק א: שאילתות SQL (40 נקודות):

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

בכל התשובות לשאלות בחלק זה:

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

1. **(q1.sql)** לכל אזור גיאוגרפי החזר את מספר המדינות בו. לעמודת מספר המדינות יש לקרוא בשם countryCount. יש להחזיר טבלה עם העמודות region, countryCount ממויינת לפי region.

```
select region, count(distinct country) as countryCount
from institutions
group by region
order by region;
```

2. **(q2.sql)** לכל אזור גיאוגרפי החזר את המספר הממוצע של מוסדות למדינה בו (כלומר אם יש באזור גיאוגרפי מסוים 2 מדינות ו-6 מוסדות נקבל 3 מוסדות בממוצע למדינה באזור זה). לעמודת הממוצע יש לקרוא בשם insAvg. יש להחזיר טבלה עם העמודות region, insAvg ממויינת לפי region.

```
select region, (count(distinct institution) + 0.0) / count(distinct country) as insAvg
from institutions
group by region
order by region;
```

3. **(q3.sql)** נאמר שמחבר הוא מומחה בתחום systems אם פרסם לפחות שני מאמרים (על פי העמודה count) בכנסים (אחד או יותר) בתחום זה. נאמר שמומחה ל-system הוא עדכני אם הוא פרסם לפחות מאמר אחד בתחום ה-systems החל מ-2014. החזר את שמות כל המומחים העדכניים בתחום ה-systems. יש להחזיר טבלה עם העמודה name ממויינת.

```
select name
from authors natural join conferences
where area = 'systems'
group by name
having sum(count) ≥ 2 and max(year) >= 2014
order by name;
```

4. **(q4.sql)** לכל אזור גיאוגרפי החזר את המדינה ממנה פורסמו הכי הרבה מאמרים באותו אזור (על פי count), ואת מספר המאמרים שפורסמו בה (אם ישנן כמה מדינות מהם פורסם אותו מספר מקסימלי החזר את כולן). יש להחזיר טבלה עם העמודות region, country, totalCount ממויינת לפי region, ואז country. הצעה – היעזרו ב-with על מנת להגדיר טבלת עזר ובה מספר המוסדות שנמצאים בכל מדינה, והאזור הגיאוגרפי בו היא שוכנת.

```
with countriesCount(region, country, totalCount) as (
select region, country, sum(count) as totalCount
from institutions natural join authors
group by region, country)

select region, country, totalCount
from countriesCount CC1
where totalCount >= ALL (
        select totalCount
        from countriesCount CC2
        where CC1.region = CC2.region)
order by region, country;
```

5. **(q5.sql)** נאמר שכנס הוא ותיק אם התקיים בלפחות עשר שנים שונות (לאו דווקא ברצף). החזר את שמות המחברים שהשתתפו רק בכנסים ותיקים (גם אם הכנס טרם היה ותיק כשפרסמו בו מאמר). יש להחזיר טבלה עם עמודה בודדת ממויינת של שמות המחברים הנקראת name.

```
with NewConferences(conference) as (
select conference
from authors
group by conference
having count(distinct year) < 10
)

select name
from authors

except

select name
from authors
where conference in (select * from NewConferences)
order by name;
```

6. **(q6.sql)** נאמר שהמרחק בין שני מחברים, a1 ו-a2, הוא 1 אם שניהם פרסמו מאמר (לפחות אחד) באותו הכנס באותה שנה. בהמשך לכך, אם מחבר a3 פרסם מאמר באותו הכנס ובאותה שנה עם a2, נוכל לומר שהמחברים a1 ו-a3 הם מחברים במרחק 2. כתבו שאילתה רקורסיבית אשר מחזירה את כל המחברים במרחק עד 2 (כולל) מהמחבר "Noam Nisan" . שימו לב, כל מחבר הוא במרחק 0 מעצמו. יש להחזיר טבלה עם עמודה אחת ממויינת בשם name ובה שמות המחברים.

```
with recursive AuthorNum(name, num) as(
select distinct name,0 from authors where name='Noam Nisan'

Union

select a1.name, num+1
from authors a1, authors a2 natural join AuthorNum AN
where a1.conference=a2.conference and a1.year=a2.year and AN.num<2)

select distinct name
from AuthorNum
order by name;
```

חלק ב: אינדקסים (60 נקודות): (להגשה בכתב בקובץ ex3.pdf)

בחלק זה של התרגיל אנחנו עדיין נשתמש בסכמה המובאת פה שוב לייתר נוחות :

authors (name, conference, year, institution, count, adjustedcount)

conferences (conference, area, subarea)

institutions (institution, region, country)

שאלה 1:

א. כתבו שאילתה בSQL המחזירה את השנים בהם פרסם מחבר מהאוניברסיטה העברית (Hebrew University of Jerusalem) מאמר, בלי כפילויות.

```
select distinct year
from authors
where institution='Hebrew University of Jerusalem';
```

ב. הריצו את השאילתה עם פקודת explain analyse, שמראה את הquery plan של השאילתה, צרפו אותה לתשובות.
כתבו כמה זמן לקח להריץ את השאילתה, והסבירו את אופן חישוב השאילתה. אם אתם לא מבינים לגמרי את ה query plan חפשו באינטרנט דוקומנטציה שתעזור לכם להסביר.

```
                                QUERY PLAN
----------------------------------------------------------------------------------------------------
-
 HashAggregate  (cost=3904.68..3905.21 rows=53 width=4) (actual time=19.308..19.343
rows=43 loops=1)
   Group Key: year
   ->  Seq Scan on authors  (cost=0.00..3902.30 rows=950 width=4) (actual time=0.453..18.548
rows=929 loops=1)
         Filter: ((institution)::text = 'Hebrew University of Jerusalem'::text)
         Rows Removed by Filter: 163895
 Planning Time: 0.109 ms
 Execution Time: 19.397 ms
```

זמן ריצה: 19.506 ms
הסבר:
השאילתה מבוצעת באמצעות מעבר סדרתי על הטבלה authors, ועל כל שורה מופעל תנאי הפילטר על פי המוסד (institution). בנוסף, יש שימוש בטבלת האש זמנית על מנת לקבץ ערכי שנה זהים יחדיו (HashAggregator).

ג. כיתבו פקודה אשר תייצר אינדקס **על שדה בודד** שישפר את זמן הריצה של השאילתה.

```
create index on authors(institution);
```

ד. הריצו את פקודת הבנייה של האינדקס ואת השאילתה עם פקודת explain analyse, שמראה את הquery plan של השאילתה, צרפו אותה לתשובות.
כיתבו כמה זמן לקח להריץ את השאילתה, והסבירו את אופן חישוב השאילתה.
אם אתם לא מבינים לגמרי את ה query plan חפשו באינטרנט דוקומנטציה שתעזור לכם להסביר.

```
QUERY PLAN
----------------------------------------------------------------------------------------------------
 HashAggregate  (cost=1617.05..1617.58 rows=53 width=4) (actual time=1.562..1.592 rows=43 loops=1)
   Group Key: year
   ->  Bitmap Heap Scan on authors  (cost=31.78..1614.68 rows=950 width=4) (actual time=0.140..0.833 rows=929 loops=1)
         Recheck Cond: ((institution)::text = institution '::text)
         Heap Blocks: exact=53
         ->  Bitmap Index Scan on authors_institution_idx  (cost=0.00..31.55 rows=950 width=0) (actual time=0.127..0.128 rows=929 loops=1)
               Index Cond: ((institution)::text = 'Hebrew University of Jerusalem'::text)
 Planning Time: 0.259 ms
 Execution Time: 1.649 ms
```

זמן ריצה: 1.808 ms

הסבר:

מעבר לאגרגציה שהוסברה קודם, השאילתה מבוצעת באמצעות אינדקס בשני שלבים.
בשלב הראשון (הפנימי) משתמשים באינדקס כדי למצוא את הכתובות של כל השורות בהן מתקיים institution ='Hebrew University of Jerusalem'. בשלב השני מביאים את השורות עצמן מהטבלה.
לאחר מציאת הכתובת של כל השורות בהן מתקיים התנאי, הבלוקים הרלוונטים מובאים מהטבלה בעזרת Bitmap Heap Scan כלומר לפני הקריאה מסדרים בעזרת Bitmap איזה שורות מכל בלוק צריך, ואז כל בלוק נקרא רק פעם אחת, והשורות שנמצאו באינדקס מוצאות לפלט.
לפעמים הBitmap לא מקדיש ביט ייחודי לכל שורה, אלא לכל בלוק, ואז התנאי נבדק שוב כנגד כל שורה בבלוק שהובא (Recheck Cond:) לפני שמועבר למשתמש.

שאלה 2:

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

הנחות:

- גודל בלוק הוא 2,000 בייטים.
- בטבלה authors יש 12,000 שורות,
- כל שורה תופסת 180 בייטים.
- התכונה conference תופסת 6 בייט.
- התכונה name תופסת 16 בייט.
- התכונה count תופסת 8 בייט.
- התכונה year תופסת 4 בייט.
- מצביע תופס 8 בייט.
- הערכים ב-count בטבלה authors הם מספרים שלמים המתפלגים אחיד בטווח [1,20]
- הערכים ב-conference בטבלה מחולקים ל80 קטגוריות באופן אחיד.

א. נתונה השאילתה הבאה, אשר הפלט עבורה הוא yes כמספר המחברים שפרסמו שני מאמרים בכנס ובשנה כלשהם:

```
SELECT "yes"
FROM authors
WHERE count = 2
```

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

בכל בלוק נכנסות $\lfloor 2{,}000 / 180 \rfloor = 11$ שורות.

הטבלה תופסת $\lceil 12{,}000/11 \rceil = 1091$ בלוקים.

עלות קריאת הטבלה כולה הוא **1,091**.

כעת, נתון האינדקס הבא על הטבלה:

```
CREATE index on authors(count)
```

2. מה תהיה דרגת הפיצול האופטימלית של האינדקס?

כל ערך באינדקס תופס 8 בייט, וכל מצביע תופס גם 8 בייט, גודל בלוק הוא 2000 בייט.

לפי הנוסחה: $d \leq \frac{b+v}{v+p} = \frac{2000+8}{8+8} \rightarrow d = 125$

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

INDEX RANGE SCAN

גובה העץ $h = \left\lceil log_{\lceil \frac{d}{2} \rceil}(T(authors)) \right\rceil = \left\lceil log_{\lceil \frac{125}{2} \rceil}(12{,}000) \right\rceil = 3$.

כמה ערכים עם count = 2: $\frac{1}{20} \times 12{,}000 = 600$ – מחזירים *yes* עבור כל ערך מתאים.

בכל עלה נכנסים לכל הפחות $\left\lceil \frac{125}{2} \right\rceil - 1 = 62$ ערכים, וסהכ צריך לעבור על $\left\lceil \frac{600}{62} \right\rceil = 10$ עלים.

אין צורך לקרוא נתונים נוספים מהטבלה.

סה"כ עלות חישוב עם אינקדס $3 + 10 = \mathbf{13}$

ב. נתונה השאילתה הבאה:

```
SELECT name
FROM authors
WHERE count = 2
```

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

כמו בסעיף א **1,091**.

כעת, נתון האינדקס הבא על הטבלה:

```
CREATE index on authors(count)
```

2. מה תהיה דרגת הפיצול האופטימלית של האינדקס?

כמו בסעיף א: $d = 125 \leftarrow \mathrm{d} \leq \frac{b+v}{v+p}$

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

INDEX RANGE SCAN+TABLE ACCESS BY ROWID

אותו אינדקס כמו א לכן $d = 125, h = 3$

שוב נדרש לעבור על 600 ערכים שונים ב-10 עלים.

כעת צריך לקרוא מהטבלה 600 שורות (בגלל ה-name), הנמצאים בלכל היותר 600 בלוקים.

סה"כ עלות חישוב עם אינקדס $\underline{3 + 10 + 600 = \mathbf{613}}$

ג. נתונה השאילתה הבאה:

```
SELECT "yes"
FROM authors
WHERE name='x' and conference='y' and year=1999
```

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

כמו בסעיף א $\underline{\mathbf{1,091}}$.

כעת, נתון האינדקס הבא על הטבלה :

```
CREATE index on authors(name, conference, year)
```

2. מה תהיה דרגת הפיצול האופטימלית של האינדקס?

כל ערך באינדקס תופס 26 בייט (עבור שלוש השדות יחד), וכל מצביע תופס 8 בייט, גודל בלוק הוא 2000 בייט.

לפי הנוסחה: $d = 59 \leftarrow \frac{b+v}{v+p} = \frac{2000+26}{26+8} \geq \mathrm{d}$

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

INDEX UNIQUE SCAN

גובה העץ $\left\lceil log_{\left\lceil \frac{d}{2} \right\rceil}(T(authors)) \right\rceil = \left\lceil log_{\left\lceil \frac{59}{2} \right\rceil}(12{,}000) \right\rceil = 3$ =h.

כיוון שמדובר במפתח יש לגשת רק לעלה אחד ובנוסף אין צורך לגשת לבלוקים של הטבלה, כי כל המידע הדרוש נמצא באינדקס.

סה"כ עלות חישוב עם אינקדס $\underline{3 + 1 = \mathbf{4}}$

ד. נתונה השאילתה הבאה:

```
SELECT name
FROM authors
WHERE conference = ‘x’ or conference = 'y'
```

כעת, נתון האינדקס הבא על הטבלה:

```
create index on authors(conference, name)
```

1. מה תהיה דרגת הפיצול האופטימלית של האינדקס?

כל ערך חיפוש בקודקוד תופס 22 בייט (עבור שני השדות), וכל מצביע תופס 8 בייט, גודל בלוק הוא 2000 בייט.

לפי הנוסחה: $d = 67 \leftarrow \frac{b+v}{v+p} = \frac{2000+22}{22+8} \geq \mathrm{d}$

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

INDEX RANGE SCAN

גובה העץ $h=\left\lceil log_{\left\lceil \frac{67}{2} \right\rceil}(12{,}000) \right\rceil = 3$. נשים לב שיש למצוא בעץ כל אחת משתי הועידות (תנאי or) ולכן יש לרדת בעץ האינדקס פעמיים.

מספר הערכים שמתאימים לשתי אפשרויות ה-conference הוא $\frac{2}{80} \times 12{,}000 = 300$

בכל עלה נכנסים לכל הפחות $\left\lceil \frac{67}{2} \right\rceil - 1 = 33$ ערכים, וסהכ צריך לעבור על $\left\lceil \frac{300}{33} \right\rceil = 10$ עלים.

אין צורך לקרוא בלוקים מהטבלה כיוון שהשמות נמצאים גם כן באינדקס.

סה"כ עלות חישוב עם אינקדס $\underline{2 \text{ x } 3 + 10 = \mathbf{16}}$

ה. נתונה השאילתה הבאה:

```
SELECT name
FROM authors
WHERE count > 1
```

כעת, נתון האינדקס הבא על הטבלה:

```
create index on authors(count)
```

1. מה תהיה דרגת הפיצול האופטימלית של האינדקס?

כמו בסעיף א: $d \leq \frac{b+v}{v+p} \rightarrow d = 125$

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

INDEX RANGE SCAN+ TABLE ACCESS BY ROWID

אותו אינדקס כמו א לכן $d = 125, h = 3$

מספר הערכים שמתאימים ל count > 1 הוא $\frac{20-1}{20} \times 12{,}000 = 11{,}400$

בכל עלה נכנסים לכל הפחות $\left\lceil \frac{125}{2} \right\rceil - 1 = 62$ ערכים, וסהכ צריך לעבור על $\left\lceil \frac{11{,}400}{62} \right\rceil = 184$ עלים.

צריך לקרוא מהטבלה 11,400 ערכים, שנמצאים בלכל היותר 1,091 בלוקים, כמספר הבלוקים בטבלה

סה"כ עלות חישוב עם אינקדס $\underline{3 + 184 + 1091 = \mathbf{1,278}}$

בהצלחה!