

# מערכות מסדי נתונים 236363 – חורף תשע"ז

## תרגיל בית רטוב מס' 2

### נושאי התרגיל: SQL

ההגשה עד: 18/12/16

בשאלות נא לפנות אל [fayeznas@gmail.com](mailto:fayeznas@gmail.com) עם נושא המכתב SQL.

בתרגיל זה תממשו תוכנית שתעבוד מול מסד נתונים של רשת חברתית המתואר ע"י הרלציות המפורטות למטה. שימו לב כי מפתחות מסומנים בקו תחתון.

Users(ID, Name, Age)

מתארת משתמשים ברשת החברתית. לכל משתמש יש מספר זהות ייחודי (ID) מסוג integer, שם (Name) מסוג text המורכב משרשור של שם פרטי ושם משפחה וגיל (Age) מסוג integer.

Follows(ID1, ID2)

מתארת את קשר המעקב - רשומה ברלציה מעידה על כך שמשתמש בעל מספר זהות ID1 עוקב אחר משתמש בעל מספר זהות ID2. קשר מעקב מאופיין ע"י שני מספרי הזהות של המשתמשים המעורבים. שימו לב שהיחס Follows אינו סימטרי (יתכן שמשתמש 123 יעקוב אחר משתמש 456 אך ההיפך לא יהיה נכון).

כתבו תוכנית לניהול מסד הנתונים. התוכנית תמומש בשפת C ותעבוד עם ספריית ה-LIBPQ של ממשק ה-PostgreSQL.

עליכם לשים לב שהתרגיל הוא לא תרגיל ב-C ולכן עליכם לכתוב כל מה שניתן ב-SQL ורק את המעטפת שתאפשר הרצה של השאילתות ב-C. ולכן:

- אסור לכם לעבד את תוצאות השאילתות ב-C אלא להדפיס אותן כפי שהן.
- אסור לכם להעביר פלט של שאילתה אחת לשאילתה אחרת באמצעות משתנה ב-C.
- וכדומה.

### **שימוש בקוד C במקום SQL יגרור הפחתה משמעותית בנקודות!**

על התוכנית לאפשר מספר פעולות שייקראו ע"י התוכנית מ-STDIN והפלט שלהן יודפס ל-STDOUT. להלן הגדרת הפלט והקלט עבור הפעולות הנדרשות.

ניתן להניח כי הפקודות שתקבלו בקלט הן פקודות חוקיות, וכי התוכן ההתחלתי של הטבלאות מתאים לסכמות הנ"ל ואינו סותר שום דרישה מהדרישות המצוינות בפעולות להלן.

במקרה והפעולה נכשלת, אין לשנות את תוכן המסד.

הקבועים SUCCESSFUL, EMPTY, NOT\_APPLICABLE ו- ILL\_PARAMS מוגדרים בקובץ wet.h.  
 הפעולות:

1. הוספת משתמש

קלט	add_user <i>Name Age</i>
תיאור הפעולה	הפעולה תוסיף לטבלת Users משתמש חדש עם שם <i>Name</i> וגיל <i>Age</i> . ה-ID של המשתמש החדש נקבע להיות ה-ID המינימלי הקיים במערכת ועוד אחד, ובמידה שהוא קיים אז המינימום ועוד 2, וכך הלאה. למשל, אם במערכת משתמשים עם מספרי זהות 1,4,6 אז מספר הזהות של המשתמש הבא שיוכנס למערכת הוא 2. ואם במערכת משתמשים עם מספרי זהות 1,2,4,6 אז מספר הזהות של המשתמש הבא שיוכנס למערכת הוא 3. במידה ואין משתמשים במערכת מספר הזהות שיבחר הוא 1.
פלט	בשורה הראשונה יודפס: ADD USER NAME: <i>Name</i> AGE: <i>Age</i> בשורה השנייה: ADD USER ID: <i>ID</i> NAME: <i>Name</i> AGE: <i>Age</i>

2. מחיקה משתמש

קלט	remove_user <i>ID</i>
תיאור הפעולה	הפעולה תסיר את המשתמש בעל מספר הזהות <i>ID</i> מהטבלה User, הרלציה Follows תעודכן בהתאם, כלומר יש להסיר עוקבים שלא קיימים כבר. הפעולה נכשלת אם לא קיים משתמש עם מספר זהות <i>ID</i> .
פלט	בשורה הראשונה יודפס: REMOVE USER ID: <i>ID</i> בשורה השנייה: במקרה של הצלחה, יש להדפיס את הקבוע SUCCESSFUL. במקרה של כישלון, יש להדפיס את הקבוע PARAMS_ILL.

3. הוספת מעקב

קלט	follow <i>ID1 ID2</i>
תיאור הפעולה	משתמש בעל מספר הזהות <i>ID1</i> עוקב אחר משתמש אחר בעל מספר הזהות <i>ID2</i> . הפעולה תכשל אם: 1. לפחות אחד ממספרי הזהות לא קיים, או שמספרי הזהות קיימים אבל זהים. 2. שהמעקב כבר קיים.
פלט	בשורה הראשונה יודפס: FOLLOW FOLLOWER: <i>ID1</i> FOLLOWING: <i>ID2</i>

<p>בשורה השנייה:  במקרה של הצלחה, יש להדפיס את הקבוע SUCCESSFUL.  במקרה של כישלון מסוג 1, יש להדפיס את הקבוע PARAMS_ILL.  במקרה של כישלון מסוג 2, יש להדפיס את הקבוע NOT_APPLICABLE.</p>	
--	--

4. מחיקת מעקב

unfollow <i>ID1 ID2</i>	קלט
<p>הפעולה מוחקת את המעקב של המשתמש ID1 אחר המשתמש ID2.</p> <p>הפעולה נכשלת אם:</p> <p>1. אחד המשתמשים לא קיים.  2. המעקב לא קיים.</p>	תיאור הפעולה
<p>בשורה הראשונה יודפס:  UNFOLLOW FOLLOWER:<i>ID1</i> FOLLOWING:<i>ID2</i></p> <p>בשורה השנייה:  במקרה של הצלחה, יש להדפיס את הקבוע SUCCESSFUL.  במקרה של כישלון מסוג 1, יש להדפיס את הקבוע PARAMS_ILL.  במקרה של כישלון מסוג 2, יש להדפיס את הקבוע NOT_APPLICABLE.</p>	פלט

5. מספר עוקבים למשתמש

following	קלט
הפעולה מחזירה לכל משתמש את מספר העוקבים שלו.	תיאור הפעולה
אין להשתמש באופרטור union	מגבלות
<p>בשורה הראשונה יודפס:  FOLLOWING</p> <p>בשורה השנייה:  בשורות הבאות, יודפסו פרטי המשתמשים בפורמט:  ID:<i>ID</i> NAME:<i>NAME</i>, HAS <i>X</i> FOLLOWERS</p> <p>כאשר <i>X</i> הוא מספר העוקבים.  הדפסת המשתמשים תהייה בשורות נפרדות והם ימוינו בסדר <u>עולה</u> של מספר העוקבים ולאחר מכן בסדר <u>יורד</u> של מספרי הזהות.  אם התוצאה ריקה (כלומר אין משתמשים ברשת) יש להדפיס את הקבוע EMPTY.</p>	פלט

6. משתמשים פופולארים

popular <i>K</i>	קלט
------------------	-----

תיאור הפעולה	הפעולה מחזירה את כל המשתמשים להם יש <b>לפחות</b> $K$ עוקבים.
פלט	בשורה הראשונה יודפס: POPULAR K:K בשורות הבאות, יודפסו פרטי המשתמשים בפורמט: ID:ID NAME:Name IS POPULAR בשורות נפרדות וימוינו בסדר <b>יורד</b> לפי מספרי הזרות. אם התוצאה ריקה יש להדפיס את הקבוע EMPTY.

7. גילוי כוכבים

קלט	star K
תיאור הפעולה	כוכב הוא משתמש שיש לו לפחות $K$ עוקבים ומוצע גילאיהם של כל עוקביו <b>גבוה</b> <b>ממש</b> ממוצע גילאי המשתמשים אחריהם הוא עוקב או, במידה והוא לא עוקב אחרי אף אחד, מגילו שלו. על הפעולה להחזיר את כל הכוכבים ברשת החברתית.
פלט	בשורה הראשונה יודפס: Star K:K בשורות הבאות, יודפסו פרטי הכוכבים בפורמט NAME:Name IS STAR בשורות נפרדות וימוינו בסדר אלפביתי <b>עולה</b> לפי השמות. אם התוצאה ריקה יש להדפיס את הקבוע EMPTY.

8. המלצות למעקב

קלט	suggest ID
תיאור הפעולה	נגדיר את דרגת מעקב באמצעות דוגמה, אם משתמש א' עוקב של משתמש ב' אז דרגת המעקב בין משתמש א' למשתמש ב' הינה 1, אם משתמש א' עוקב של משתמש ב' ומשתמש ב' הוא עוקב של משתמש ג' אז דרגת המעקב בין משתמש א' למשתמש ג' היא 2, וכו'. אם משתמש עוקב של משתמש אחר עם יותר מדרגה אחת אז הדרגה המינימלית היא הקובעת. נגדיר שלמשתמש ID מומלץ לעקוב אחר משתמש, אם דרגת המעקב ביניהם גדולה או שווה ל-2. על הפעולה להחזיר את כל המשתמשים הממולצים למעקב של משתמש ID. הפעולה נכשלת אם המשתמש לא קיים.
פלט	בשורה הראשונה יודפס: Suggest ID:ID בשורות הבאות, במקרה של הצלחה:

אם יש המלצות יודפס: YOU MAY KNOW: <i>NAME</i> ובשורות הבאות יודפסו שמות המשתמשים המומלצים, בשורות נפרדות וימינו בסדר אלפביתי <u>עולה</u> של שמותיהם. אם התוצאה ריקה יש להדפיס את הקבוע EMPTY. במקרה של כשלון: יש להדפיס את הקבוע PARAMS_ILL.	
--	--

## הנחיות כלליות

### דרישות התוכנית:

1. התוכנית תמומש בשפת C
2. התוכנית תעבוד עם ספריית ה-LIBPQ של ממשק ה-PostgreSQL.
3. התוכנית תכתב כולה בקובץ יחיד

### התחברות למסד הנתונים:

באתר הקורס מוגדר הקובץ wet.h אשר מכיל את שתי השורות הבאות:

```
#define USERNAME "my_user"
#define PASSWORD "my_pass"
```

עליכם לשנות קובץ זה על מנת ש-`<user_my>` ו-`<pass_my>` יהוו את שם המשתמש והסיסמה שלכם במסד הנתונים (מחרוזות). כמו כן, בקובץ C המכיל את התוכנית שלכם יש לבצע:

```
#include "wet.h"
```

ולשתמש בפקודה הבאה (בדיוק) על מנת להתחבר למסד נתונים:

```
char connect_param[80];
/* ... */
sprintf(connect_param, "host=pgsql.cs.technion.ac.il dbname=%s user=%s password=%s",
USERNAME, USERNAME, PASSWORD);
conn = PQconnectdb(connect_param);
```

### הידור התוכנית:

לנוחותכם נתון קובץ Makefile באתר הקורס.

### טבלאות:

בחשבונכם יוגדרו בקרוב כל הטבלאות של מסד הנתונים עם תוכן התחלתי. טבלאות אלו הן לקריאה וכתובה. בנוסף, יוגדרו טבלאות עם קידומת `course_` אשר נועדו לקריאה בלבד (שאלתות (SELECT)).

במידה ושינייתם טבלה, נאמר `tableName`, על מנת לשחזר את התוכן המקורי שלה, יש לבצע את הפקודות הבאות:

```
DROP TABLE tableName;
CREATE TABLE tableName AS SELECT * FROM course_tableName;
```

### בדיקת התוכנית מול קבצי קלט:

על מנת לבדוק את התוכנית שלכם מול קבצי קלט של הבדיקה הראשונית, יש להוריד מהאתר את קבצי הבדיקה הראשונית ולבצע את הפקודות:

```
wet < 1.in > 1.out
diff -w 1.out 1.sol
```

הפקודה הראשונה מפעילה את התוכנית wet עם קלט מתוך הקובץ in.1, וכותבת את הפלט לקובץ 1.out. הפקודה השנייה משווה את התוצאה לפלט הרצוי שנמצא ב-1.sol. אם תוצאת ההשוואה ריקה (תקבלו את ה-prompt בחזרה מיד), ההשוואה הצליחה. שימו לב: התוצאות מתייחסות לתוכן ההתחלתי של הטבלאות.

### הנחיות תכנות:

- יש לכתוב את התוכנית בשפת C תוך שימוש בספריית LIBPQ. התרגול שסוקר את פונקציות הספרייה ומדגים את שלד התוכנית נמצא באתר הקורס (שם המצגת הינו PotgreSQL.ppt).
- ניתן להשתמש בטבלאות זמניות, אך יש למחוק אותן לפני סיום התוכנית. זכרו שתוכניתכם לא תיבדק מתוך חשבונכם, לכן אין להתייחס בקוד שלכם לאף טבלא שאינם זמניים או אינם חלק ממסד הנתונים.
- לא ניתן להשתמש במבטים.
- יש להקפיד על שחרור משאבים (connection, result, tables-temp וכד') לפני סיום התוכנית.
- יש להשתמש ב-C ANSI (ללא תוספות לא סטנדרטיות).
- יש לכתוב את הקוד בצורה מבנית וברורה, תוך הקפדה על מתן שמות משמעותיים למשתנים ולפונקציות, וכד'.
- אין להשתמש בהקצאות זיכרון דינאמיות או מערכים גדולים.
- יש לבדוק מקרי קצה ומקרי שגיאה בקפידה!
- את שורות הפלט יש להדפיס בדיוק בפורמט המצויין, עם "n" (בסוף ככל שורה ל-stdout) יש להשתמש ב-MACROS המוגדרים ב-wet.h.
- את כל הודעות השגיאה הבלתי צפויות (למשל, הקשורות לבעיות בעבודה עם מסד הנתונים) יש להדפיס ל-stderr.
- התוכנית תיבדק באופן אוטומטי על המחשב cs12 (מערכת הפעלה Linux).
- בנוסף לבדיקה מול קבצי קלט, תתבצע בדיקה לאיתור העתקות ושיכפול קוד (בין עבודות שונות) או שימוש בקוד C במקום SQL.

### הגשה אלקטרונית:

יוגש קובץ C יחיד - לא מכווץ (ZIPPED). יש להגיש את הקובץ (ללא ה-wet.h) באמצעות מנגנון ההגשה האלקטרונית באתר הקורס. ניתן לבצע את ההגשה ע"י אחד מהשותפים, אך יש לציין את מספר ת"ז של השותף השני. ניתן לבצע את ההגשה האלקטרונית יותר מפעם אחת, אך ההגשה האחרונה הקודמת למועד ההגשה היא זו שתקבע.