

## תרגיל בית מספר 1 – DDL, SQL

### הנחיות להגשת התרגיל:

1. תאריך הגשה - 15/02/24 בשעה 23:55.
2. מתרגלת אחראית על המטלה: קורל שרף
3. יש להגיש את תרגיל הבית בזוגות בלבד. ציון המטלה של סטודנט שיגיש לבד יתחיל מ-90.
4. יש להגיש תיקיית zip הכוללת שני קבצים בלבד: פתרון בקובץ sql לשאלה 1 (בנושא DDL) ופתרון לשאלות 2-4 בקובץ python (ראו הנחיות בסוף מסמך זה). שם תיקיית ה-zip צריך להיות מספרי זהות של המגישים בפורמט: ID1\_ID2 כאשר ID1, ID2 הם מספרי הזהות של הסטודנטים. שם קובץ ה-sql (שאלה 1) צריך להיות ID1\_ID2\_Q1. למשל, עבור זוג סטודנטים שמספרי הזהות שלהם הם 123456789 ו-987654321, התשובות לשאלה הראשונה יוגשו בקובץ sql בשם 123456789\_987654321\_Q1, ושם תיקיית ה-zip אותה הם יגישו יהיה 123456789\_987654321.
5. את הקובץ יש להגיש דרך אתר הקורס במקום המתאים ב-moodle על ידי **אחד** מבני הזוג.
6. איחור בהגשת התרגיל יגרור קנס בגובה 20% מהציון עבור כל יום איחור (פרט למקרים חריגים כגון מילואים או סיבה בריאותית). במקרים אלה יש לפנות למתרגל האחראי על התרגיל.
7. פתרונות לשאלות שלא יעברו את הבדיקה האוטומטית יקבלו את הציון 0. ניתן להגיש ערעור, אך בהנחה שיתברר כי הסיבה לכישלון בבדיקה היא אי מעקב אחרי הוראות ההגשה, ציון הסטודנטים יתחיל מ-90 והתרגיל ייבדק מחדש (ראו פירוט המופיע בעמודים 5-7 של מסמך זה).

## חלק א' - DDL (40 נקודות)

### הקדמה:

ברכות! מוניתם לאחראים על יצירת מסד הנתונים של רשת חברתית חדשה בהתאם לדרישות המופיעות בתיאור שלפניכם.

המערכת מתעדת את כל המשתמשים הרשומים ברשת. כל משתמש מזוהה באמצעות מזהה משתמש ייחודי הניתן לו בעת ההרשמה (מחרוזת טקסטואלית). עבור כל משתמש יש לתעד שם המשתמש שלו, הסיסמה שלו והמדינה בה הוא גר. כל משתמש מחויב בעת הרשמתו לרשת לבחור שם משתמש הכולל בדיוק 8 תווים, ויש לוודא כי אין שני משתמשים שונים או יותר שחולקים את אותו שם המשתמש.

המערכת מתעדת את כל התכנים שמשתמשים פרסמו ברשת. תוכן הוא הודעה טקסטואלית לא ריקה המפורסמת על ידי משתמש הרשום ברשת ומכילה לכל היותר 280 תווים. לכל תוכן נשמר מועד פרסום התוכן (בפורמט DATETIME) וניתן לזהות אותו לפי מועד זה ומזהה המשתמש שפרסם את התוכן. יש לתעד רק תכנים שפורסמו לאחר שנת 2019.

בנוסף, הרשת מאפשרת למשתמשים לפרסם פריטים למכירה (למשל, מוצרי חשמל, ריהוט ועוד) מבין סט סגור של מוצרים שמוגדר במסד. מוצר מזוהה על ידי שמו, ובנוסף יש לתעד את הקטגוריה אליה הוא משתייך, ואת המחיר המקסימלי שניתן לדרוש עבור המוצר (ערך מספרי חיובי רציף). שדות הקטגוריה והמחיר חייבים להיות מוזנים (כלומר אסור להשאירם ריקים).

כל משתמש שמפרסם פריט למכירה נדרש להזין את שם הפריט, מחיר מבוקש, מצב הפריט (מבין הערכים "new", "used", "fair", "broken") וקישור לתמונה של הפריט (כתובת אינטרנט המורכבת מלכל היותר 100 תווים). לא ייתכנו שני פריטים למכירה בעלי אותו השם ששייכים לאותו המשתמש. המערכת מאפשרת לצרף קישורים לתמונות מסוג PNG בלבד, ועל כן נרצה לוודא שהקישור מסתיים ב-"PNG". ושלפני סיומת זו ניתן שם לתמונה בעל תו אחד לפחות. בנוסף, הרשת מחייבת להזין מחיר גדול או שווה ל-0, שלא עולה על המחיר המקסימלי שניתן לבקש עבור המוצר. אם המשתמש לא הזין מחיר בעת פרסום המודעה, הרשת מניחה שהפריט למסירה ולכן מגדירה באופן אוטומטי את המחיר כ-0. במקרה שפריט כלשהו נמכר, נרצה לתעד מיהו המשתמש הרשום ברשת שקנה את אותו הפריט.

חלק מפרסומי המכירה כוללים מידע שגוי, למשל הזנת מצב המוצר כחדש בעוד שבתמונה הפריט נראה שבור. על כן, הרשת מאפשרת למשתמשים רשומים לדווח על פרסומים כאלו. נרצה לתעד מהו הפרסום השגוי, מי המשתמש שדיווח על המוצר ומהו תאריך הדיווח. כדי למזער את מספר דיווחי השווא, נרצה לוודא שמשתמש לא יכול לדווח על יותר מ-2 פרסומי פריטים שונים ביום. כמו כן, משתמש לא מדווח על פרסום של עצמו. לכל דיווח נשמור הכרעה בנוגע להאם הדיווח מוצדק (משתנה בינארי שמקבל את הערך 1 אם הדיווח מוצדק ו-0 אחרת).

### **שאלה 1 (40 נקודות):**

עליכם לכתוב סקריפט DDL אשר מייצר את הטבלאות בהתאם לתיאור הנתון. אם קיימת דרישה אשר לא ניתן לממש באמצעות DDL אנא ציינו זאת במפורש בקובץ ה-sql כהערה (באמצעות '--'). כל עוד לא נאמר אחרת בתיאור, ניתן להניח ששדה מסוג מחרוזת הוא באורך של לכל היותר 50 תווים.

## חלק ב' – SQL (60 נקודות):

לרשת החברתית תיאור **חלקי** של מסד נתונים המתעד מידע אודות משתמשים, מעקב של משתמשים אחרי משתמשים אחרים ואינטראקציות של משתמשים עם תכנים. שימו לב שייתכן שהרלציות לא תואמות את הדרישות המתוארות בחלק א'. עבור כל אחת מהרלציות, ה-**primary key** מסומן באמצעות קו תחתון. להלן תיאור הרלציות:

**Users(ID, name, cName)** - רלציה המתעדת את כלל המשתמשים המופיעים במערכת.

ID - מזהה המשתמש.

name - שם המשתמש.

cName - שם המדינה ממנה בה גר המשתמש.

**Follows(ID1, ID2)** - רלציה המתעדת מעקב של משתמש אחרי משתמש אחר.

ID1 - מזהה המשתמש של העוקב (מפתח זר לרלציית Users).

ID2 - מזהה המשתמש של הנעקב (מפתח זר לרלציית Users).

**Interactions (authorID, cNum, uID, iType)** - רלציה המתעדת את האינטראקציות של משתמשים עם

תכנים, כאשר משתמש יכול לאהוב תוכן, לעקוב אחריו, להגיב עליו ולשתף אותו.

authorID - מזהה כותב התוכן (מפתח זר לרלציית Users).

cNum - מספר סידורי של התוכן. התכנים שכותב תוכן מפרסם ממוספרים.

uID - מזהה המשתמש שהיה באינטראקציה עם התוכן (מפתח זר לרלציית Users).

iType - סוג האינטראקציה, מבין הסוגים "like", "follow", "comment", "share".

עליכם לענות על שאילתות ה-SQL הבאות תוך שימוש בטבלאות הנתונות בחלק ב' בלבד.

### **שאלה 2 (20 נקודות):**

כתבו שאילתה אשר מוצאת עבור כל מדינה שגרים בה לפחות 30 משתמשים, את מספר המשתמשים שכתבו תוכן שלפחות משתמש אחד שיתף ("share"). החזירו את התוצאה בצורה ממוינת בסדר יורד לפי מספר המשתמשים. במקרה של שוויון במספר המשתמשים בין מדינות שונות, יש להדפיסם בסדר עולה לפי שם המדינה.

### **שאלה 3 (20 נקודות):**

אינטראקציה שגויה היא מצב בו משתמש היה באינטראקציה עם תוכן שנכתב על ידי משתמש שהוא לא עוקב אחריו. עבור כל המשתמשים שאף פעם לא כתבו תוכן שקיימת עבורו אינטראקציה שגויה, החזירו את מזהה המשתמש ואת מספר התכנים אותם הוא רשם. החזירו את התוצאה בצורה ממוינת בסדר עולה לפי מזהה המשתמש. הערה: יש להחזיר רק את המשתמשים שמקיימים את התנאי באופן לא ריק, כלומר רק משתמשים שמקיימים את התנאי בשאלה ובנוסף קיים להם לפחות תוכן אחד עם אינטראקציה.

### **שאלה 4 (20 נקודות):**

משתמש בינלאומי הוא משתמש שקיים לו עוקב אחד לפחות מכל מדינה שמתועדת במסד הנתונים. נגדיר פעילות בינלאומית של משתמש כמספר האינטראקציות שמשתמש ביצע אשר מקיימות את התנאי הבא: המשתמש שפרסם את התוכן גר במדינה שונה מהמדינה בה גר המשתמש שהיה באינטראקציה עם התוכן. עבור כל משתמש בינלאומי שהפעילות הבינלאומית שלהם היא לפחות 65, יש להחזיר את מזהה המשתמש, את שמו ואת מספר העוקבים שלו. הערה: אם משתמש היה באינטראקציה מסוג "like" וגם אינטראקציה מסוג "share" עם תוכן בעל מספר סידורי 2 שכתב כותב עם מזהה 123, מספר האינטראקציות שיספרו עבור הפעילות הבינלאומית שווה ל-2 (כל עוד התנאי מתקיים).

**הערה חשובה-** אסור להשתמש בשלב זה של הקורס בכלים כמו Views או With לצורך יצירת טבלאות ביניים. רק בהמשך הקורס נלמד כיצד לעשות זאת.

**בהצלחה!**

## הנחיות הגשה

באחריותכם לוודא שמבנה הרלציה (כולל סדר השדות) בשאילתות שתגישו תואם את המבנה הנדרש כפי שמופיע בפלטים לדוגמה (יש משמעות לסדר התכונות בתוצאה).

הבדיקה בחלק זה תורץ באופן אוטומטי ותתבצע עבור מספר קלטים שונים. לתרגיל מצורף קובץ בשם `create_table_commands_PartB.sql` שכולל פקודות DDL ליצירת הטבלאות הנתונות. בנוסף, מצורפים קבצי קלט ופלט לדוגמה. תיאור הקלטים, אופן העלאת הנתונים לטבלאות והפלטים הנכונים מצורפים בנספח 2 שבסוף המסמך.

מומלץ ליצור את הטבלאות במסד הנתונים שלכם ולבדוק את נכונות הפתרון מול הקלטים והפלטים הנתונים בעצמכם ורק אז לוודא את תקינות ההגשה במנגנון הבדיקה האוטומטית (מוסבר עוד מעט). שימו לב כי בבדיקה האוטומטית שנריץ אצלנו נשתמש בקלטים ופלטים נוספים ולכן אנו ממליצים לכם לחשוב על מקרים נוספים ולבחון את עצמכם עליהם, שכן, ייתכן כי הפתרון שלכם יחזיר תוצאה נכונה עבור קלט מסוים אך לא עבור קלטים אחרים.

לתרגיל מצורף קובץ בשם `Automatic_Grader.py` אשר מריץ בדיקה אוטומטית על קבצי הקלט והפלט. עליכם לבצע מספר פעולות כדי להריץ את הבדיקה.

1) הזינו את מספרי הזהות של שני השותפים בפורמט `ID1_ID2` תחת המשתנה `IDS` שבשורה 242. יש להקפיד על סדר כתיבת מספרי הזהות כמו בשם קובץ ה-`zip` אותו אתם מגישים. למשל, עבור זוג סטודנטים שמספרי הזהות שלהם הם 123456789 ו-987654321, יש לכתוב בשורה 242 (שימו לב כי קו תחתון מפריד בין המספרים):

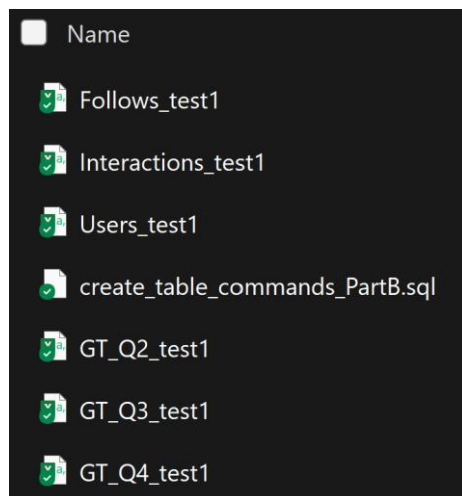
```
IDS = "123456789 987654321"
```

2) שנו את שם את הקובץ `ID1_ID2_Queries.py` בהתאם למספרי הזהות שלכם. למשל, עבור זוג סטודנטים שמספרי הזהות שלהם הם 123456789 ו-987654321 יש לקרוא לקובץ `123456789_987654321_Queries.py`. בקובץ זה תכתבו את השאילתות שלכם ואותו תגישו. **אין צורך להגיש גם את הקובץ `Automatic_Grader.py`.**

3) הזינו את שם המשתמש הטכניוני של אחד מבני הזוג תחת השדות `DATABASE_NAME` ו-`USER_NAME` בשורות 245 ו-246 (לא לשכוח להחליף "0-ב" במקרה הצורך).

```
SERVER_NAME = "techniondbcourse01.database.windows.net"
DATABASE_NAME = "user"
USER_NAME = "user"
PASSWORD = "Qwerty12!"
```

4) ודאו כי הקבצים המצורפים מופיעים בתיקיה בשם data, אשר יושב בתיקיה המרכזית בה ממוקם הקובץ Automatic\_Checker.py. יש לעשות לתיקיה unzip ולשים אותה בתיקיה רגילה עם כל הקבצים (ראו תמונה בנקודה 6 שבעמוד הבא) תוכן התיקיה הוא:

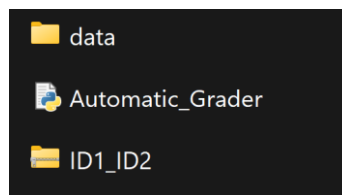


5) בקובץ ששמו (לפני שתשנו אותו בהתאם למספרי הזהות שלכם) ID1\_ID2\_Queries.py מוגדר מילון בשם QUERY\_ANSWERS. למילון זה שלושה מפתחות – Q2, Q3 ו-Q4. הערך אותו יש להזין עבור כל אחד ממפתחות אלו הוא השאילתה המתאימה. אנו ממליצים לכם לעבוד קודם כל דרך ה-console, לוודא שאתם בטוחים לגבי תשובתכם ורק לאחר מכן להעתיק את השאילתה למקום המתאים ולהריץ את Automatic\_Grader.py. עליכם לכתוב את השאילתות שלכם בין המרכאות הכפולות (במקום תבנית השאילתה שמכילה את הפקודות SELECT, FROM ו-WHERE):

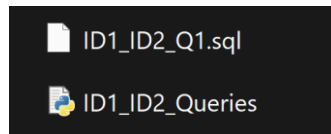
```
QUERY_ANSWERS = {
    "Q2":
        """
        SELECT
        FROM
        WHERE;
        """
    ,
    "Q3":
        """
        SELECT
        FROM
        WHERE;
        """
    ,
    "Q4":
        """
        SELECT
        FROM
        WHERE;
        """
}
```

מדגישים שוב – זהו הקובץ שעליכם להגיש (לצד הפתרון של שאלה 1 בנושא DDL).

6) התיקיה שבה נמצא קובץ ההרצאה האוטומטית צריכה להכיל את הקבצים הבאים (עם החלפת השם ID1\_ID2 במספרי הזהות שלכם כמובן):



כאשר תיקיית ה-zip היא התיקיה אותה אתם מגישים שמכילה את הקבצים הבאים (עם החלפת השם ID1\_ID2 במספרי הזהות שלכם כמובן):



7) כעת, הריצו את הקובץ Automatic\_Grader.py. אם אכן פתרונכם נכון עבור הקלט הנתון, הפלט אותו תראו יהיה:

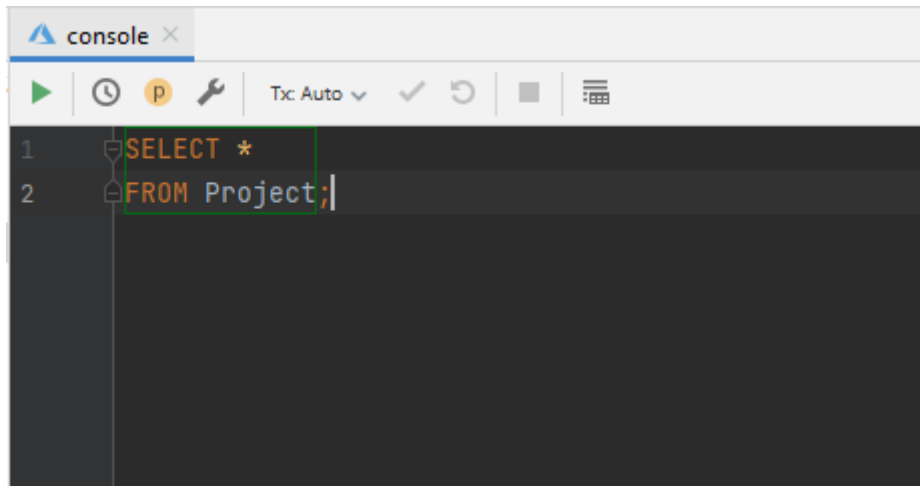
```
*****
query 2, test 1: Good
*****
query 3, test 1: Good
*****
query 4, test 1: Good
*****
```

**הערה חשובה -** פתרונכם ייבדק בצורה אוטומטית. **באחריותכם** לוודא כי הקפדתם על הנחיות ההגשה המפורטות בקובץ זה. אם הבדיקה האוטומטית נכשלה ציונכם בתרגיל יהיה 0. כמובן שתוכלו להגיש ערעור והתרגיל ייבדק מחדש אך ציונכם יתחיל מ-90 (אלא אם יתברר שלא חרגתם מאף אחת מההנחיות). אנא ודאו כי ההגשה שלכם מקיימת את כל סעיפי ה-checklist הבא:

- הגשת קובץ SQL שפורמט שמו ID1\_ID2\_Q1 (יש להחליף את ID1 ו-ID2 במספרי הזהות של הסטודנטים המגישים) עבור פקודות ה-DDL המבוקשות בשאלה 1.
- לצורך הבדיקה – הזנת מספרי הזהות של המגישים בשורה 242 בקובץ Automatic\_Checker.py.
- לצורך הבדיקה – הזנת שם המשתמש של אחד מבני הזוג בשורות 245 ו-246 בקובץ Automatic\_Checker.py.
- הגשת הקובץ שפורמט שמו הוא ID1\_ID2\_Queries.py (יש להחליף את ID1 ו-ID2 במספרי הזהות של המגישים) כאשר במילון המוגדר בקובץ זה יש להזין את פתרונותיכם לשאלות 2-4.
- יש לשמור על הסדר הפנימי בין ID1 ל-ID2 בכל המקומות בהם אתם נדרשים לכתוב את מספר הזהות. למשל אין לכתוב בשורה 202 בקובץ 987654321\_123456789 ולהגיש את התרגיל כולו תחת השם 123456789\_987654321.

## נספח 1 – שמירת תוכן שנכתב ב-console

נניח כי ברצוננו לשמור כקובץ sql את השאילתה הבאה אותה כתבנו ב-console :



The screenshot shows a SQL console window with a dark background. The title bar says 'console'. Below the title bar is a toolbar with icons for running, undo, redo, and other functions. The main area shows two lines of SQL code: '1 SELECT \*' and '2 FROM Project;'. The second line is highlighted with a green box.

(1) לחיצה עם הכפתור הימני של העכבר על כותרת הלשונית (ליד הלוגו הכחול).

(2) ריחוף על האפשרות של Open in.

(3) בחירה באפשרות של Explorer (במחשבי מאק הפקודה היא Finder)

(4) כעת תיפתח התיקיה בה נשמר הקובץ. שנו את שמו בהתאם לצרכיכם.

## נספח 2 – העלאת רשומות לרלציה מקובץ CSV

1. צרו את הטבלאות במסד הנתונים על ידי הרצת הפקודות בקובץ create\_table\_commands\_PartB.sql מתוך ה-console.

2. הזינו את הערכים המופיעים בקבצי ה-CSV של הטסטים באמצעות טעינה ישירה שלהם למסד הנתונים. עליכם לעשות זאת בהתאם לסדר יצירת הטבלאות בקובץ יצירת הטבלאות.

ניתן לעשות זאת באמצעות באופן הבא :

א) לחיצה (עם הלחצן הימני של העכבר) על שם הרלציה, בחירה באפשרות של Import/Export ובה באפשרות Import Data from file.

ב) בחירת קובץ ה-csv הרלוונטי מהתיקייה בה הוא יושב על מחשבכם.

ג) וידוא המיפוי של השדות המופיעים בקובץ לשדות ברלציה.

ד) לחיצה על OK ווידוא קבלת הודעת הצלחה על העלאת כל הרשומות.

3. לאחר שטענתם את כל הרלציות ניתן להתחיל בעבודה.