תרגיל 1 : מידול דיאגרמות ישויות קשרים

תאריך הגשה : 06/04/2025 ,23:55

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

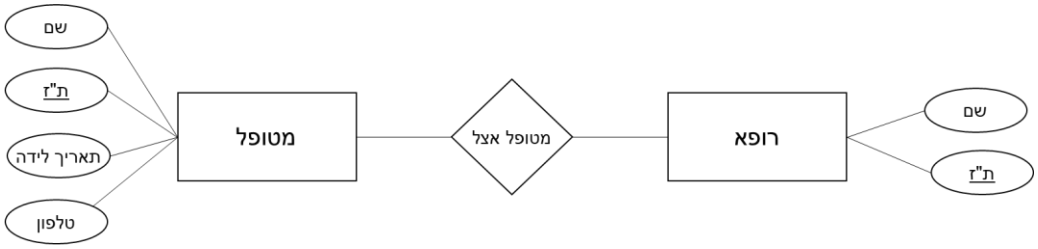
- ex1.pdf עם התשובות לשאלות להלן.
- create.sql
- drop.sql
- ex1.py
- README שמכיל שורה בודדת ובו ה-login של הסטודנט שמגיש את התרגיל. אם התרגיל מוגש בזוגות, על שורה זאת להכיל את שני ה-login מופרדים בפסיק.

שימו לב:

- התרגיל צריך להיות מוגש כPDF מוקלד.
- את החלקים בתרגיל שבהם אתם נדרשים לצייר דיאגרמות, תוכלו לצייר באופן ידני, לסרוק באיכות טובה, ולהדביק במקומות המתאימים בתוך הPDF של הפתרון.

<u>שאלה 1:</u>

נתונה דיאגרמה בסיסית של מסד נתונים שמכיל מידע אודות רופאים ומטופלים במרפאה.
לכל מטופל יש שם, מספר ת"ז, תאריך לידה וטלפון.
לכל רופא יש שם ות"ז.
במסד נשמר גם המידע על איזה רופא מטפל באיזה מטופלים



בכל סעיף יש לצייר בדיאגרמה רק את המידע הנדרש באותו סעיף. (בכל סעיף יש לצייר מחדש את הדיאגרמה)

<u>הערה:</u> אם יש אילוצים בשאלה שאתם לא מצליחים לאכוף בדיאגרמה, תוכלו לצרף הסבר למה אילוצים אלו לא נכנסו.

א) איך היית משנה את הדיאגרמה הבסיסית אם ידוע שכל מטופל מקבל טיפול אצל בדיוק רופא אחד?

ב) איך היית משנה את הדיאגרמה הבסיסית אם ידוע שחלק מהרופאים צריכים הדרכה, והם מוצמדים לרופא אחר יחיד שידריך אותם.

ג) איך היית משנה את הדיאגרמה הבסיסית אם ידוע שהמרפאה מקבלת מטופלים ששייכים ל3 קופות חולים : 'מצויינת', 'מושלמת' ו'מעולית'. כל מטופל במרפאה שייך לקופת חולים אחד בדיוק ויש לשמור את תאריך ההצטרפות שלו לקופה.

ד) איך היית משנה את הדיאגרמה הבסיסית אם ידוע שיש שני סוגי רופאים במרפאה, רופאי משפחה ורופאים מומחים. לרופא מומחה נרצה לשמור את תחום המומחיות שלו.
בנוסף, חלק מהמטופלים במרפאה מוגדרים כמטופלים בעלי סיכון גבוה, ועבורם נרצה לשמור מספר המציין את רמת הסיכון שלהם. מטופלים בסיכון יכולים להיות מטופלים אצל רופא משפחה או אצל רופא מומחה, אבל מטופלים רגילים יכולים להיות מטופלים רק אצל רופא משפחה.

ה) איך היית משנה את הדיאגרמה הבסיסית אם ידוע שהמרפאה מעוניינת לשמור את הפרטים עבור כל טיפול שמטופל קיבל אצל רופא במרפאה ולא רק סתם לשמור איזה מטופל שייך לאיזה רופא. עבור כל טיפול של מטופל אצל רופא רוצים לשמור מי המטופל ומי הרופא המטפל, וכמו כן, את מספר הטיפול (ראשון/שני/שלישי/וכו...), תאריך הטיפול, ואת הסיכום של הטיפול.

שאלה 2:

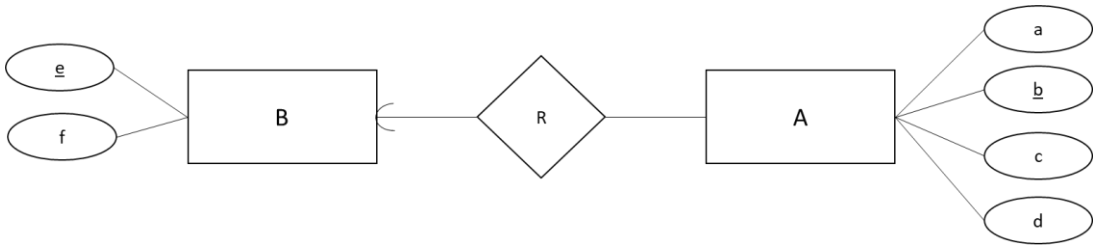
בכל סעיף:

i) יש לתרגם את הדיאגרמה ליחסים ולציין את השדות של כל יחס, ואת המפתחות. אם יש כמה אפשרויות למפתח, ציינו את כולן. אם יש ירושה (isA), תרגמו בשיטת E/R style.
הערה: ייתכן ולא ניתן יהיה לבטא את כל האילוצים של הדיאגרמה בתרגום היחסים. אם יש אילוצים שלא באו ליידי ביטוי בתרגום, ציינו אותם בתשובה. אם יש דרכים שונות לתרגם כך שבכל תרגום יבואו לידי ביטוי אילוצים אחרים, ציינו את כל הדרכים לתרגום ואילו אילוצים כל אחד משמר.

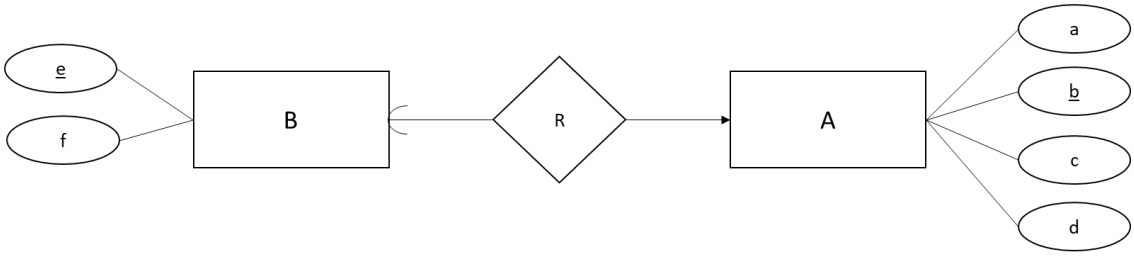
ii) נסמן ב-|A| את מספר הישויות בקבוצת הישויות A. מה ניתן לומר על מספר הישויות בקבוצה A לעומת מספר הישויות בקבוצה B ?
יש להתייחס לשתי קבוצות אלו בלבד ולהשתמש בסימנים: >, <, =>, =<, =. במקרה שלא ניתן לקבוע יש לציין "לא ניתן לקבוע".

הערה: ניתן להניח שאף קבוצת ישויות לא ריקה.

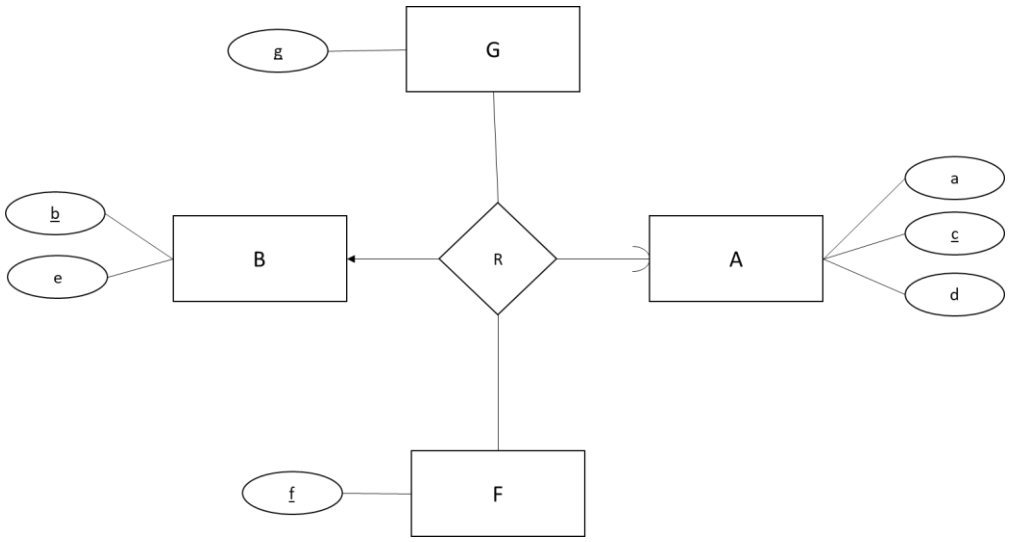
א)



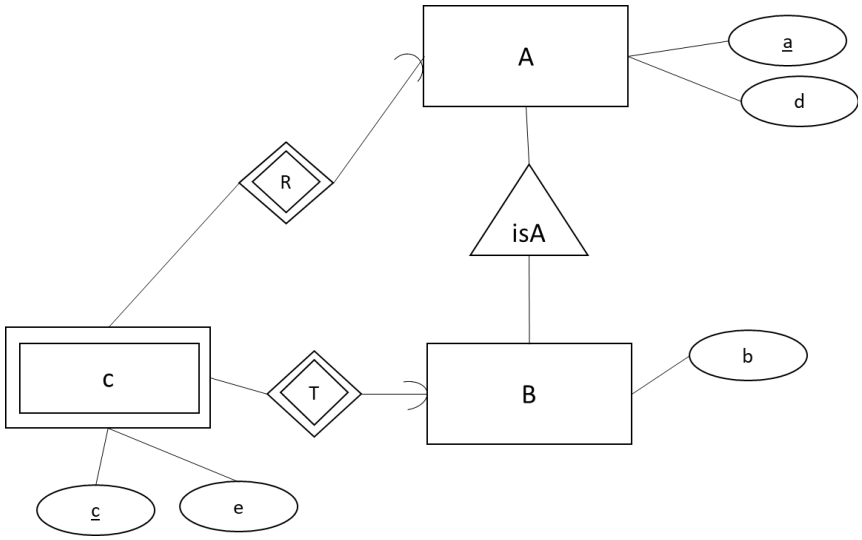
ב)



ג)



ד)



שאלה 3:

אתם נדרשים לתכנן ולבנות מסד נתונים עבור מאגר מידע אמיתי של נתוני קבלה לאוניברסיטאות ברחבי העולם. כדי להשיג את המידע השתמשנו באתר http://www.kaggle.com. האתר Kaggle מהווה קהילה למדעני דאטה ולמידה חישובית. היא מאפשרת למשתמשים למצוא ולפרסם מאגרי מידע, לבנות מודלים חישוביים ולעבוד עם מומחים אחרים כדי לגלות תובנות. מכיוון שכך, זה אתר שחשוב להכיר.
לצורך התרגיל שלנו, אנחנו לא נשתמש במידע הגולמי כמו שהוא הופיע באתר של Kaggle, מכיוון שכמו בהרבה מקרים, המידע באתר מכיל כל מיני טעויות שיכולות להקשות על ביצוע התרגיל. אנחנו נספק לכם העתק שעבר preprocess לצורך ניקוי של מידע שגוי. בנוסף, בקובץ שאנחנו נספק לכם, מופיעות רק חלק מהעמודות שמופיעות במידע במקורי כדי לפשט את הפתרון.

תוכלו למצוא את הקובץ מכווץ בפורמט zip באתר של הקורס, וגם במערכת המחשבים במעבדה בתיקיה:
~db/data/ex1/enrollment.zip

ניתן להעתיק אותו לתיקיה שלכם.

הקובץ הזה מכיל טבלה אחת ענקית עם כל המידע על נתוני קבלה לאוניברסיטאות שונות בין השנים 1950-2020. בטבלה הנתונה מופיעות העמודות הבאות:

country	שם המדינה בה ממוקמת האוניברסיטה
countrycode	קוד המדינה בן 3 אותיות
region	אזור גיאוגרפי – יש 7 אזורים גיאוגרפיים
incomegroup	קבוצת ההכנסה של המדינה – יש 4 קבוצות הכנסה
iau_id1	מזהה ייחודי של האוניברסיטה
eng_name	שם באנגלית של האוניברסיטה
orig_name	השם המקורי של האוניברסיטה
foundedyr	שנת ההקמה של האוניברסיטה
yrclosed	שנת הסגירה של האוניברסיטה- מופיע רק אם האוניברסיטה נסגרה
private01	ערך בינארי האם האוניברסיטה פרטית
latitude	קו רוחב של הקואורדינטות גיאוגרפיות של האוניברסיטה
longitude	קו אורך של הקואורדינטות גיאוגרפיות של האוניברסיטה
phd_granting	ערך בינארי האם האוניברסיטה מעניקה תואר PhD
divisions	מספר המחלקות באוניברסיטה
specialized	ערך בינארי האם האוניברסיטה מעניקה תואר בתוכניות מיוחדות
year	השנה עבורה יש ערך של נתוני קבלה
students5_estimated	הערכה של מספר הסטודנטים שהתקבלו לאוניברסיטה באותה השנה

שימו לב : לצורך בניית הדיאגרמה (סעיף א ותרגום בסעיף ב) ניתן להניח שכל המידע שאמור להופיע בטבלה קיים למרות שבנתונים עצמם חסר מידע (ועם זה תצטרכו להתמודד בסעיף ד).

במסד נתונים אמיתי לא כדאי לשמור את המידע בצורה כזאת, כי יש בו הרבה מאוד כפילות מידע.
בתרגיל זה אנחנו ננסה למדל את הנתונים בצורה נכונה בעזרת דיאגרמת ER, ובהמשך לטעון את הנתונים לתוך הטבלאות הנגזרות מהדיאגרמה.

א) ציירו דיאגרמת ישויות קשרים מתאימה הממדלת את המידע בעמודות של הקובץ enrollment.csv. מומלץ להוסיף תיאור מילולי של הדיאגרמה המכיל את כל הידע. אם הסתמכתם על הנחות שלא נאמרו במפורש, חובה לציין אותן. ייתכן שבדיאגרמה לא תצליחו למדל את כל ההנחות שמתקיימות בנתונים. במקרה כזה, ציינו אילו הנחות הדיאגרמה שלכם איננה ממדלת.

ב) תרגמו את כל הדיאגרמה ליחסים רלציונים. לכל יחס ציינו את האטריביוט/ים שהם המפתח.
אם יש מספר אפשרויות למפתח מספיק לבחור מפתח אחד.

את סעיפים א׳ וב׳ יש להגיש בקובץ ex1.pdf ביחד עם התשובות לשאלות 1 ו-2.

בחלק הבא תשתמשו במסד הנתונים Postgres ובקוד python כדי לבנות טבלאות ולטעון את הנתונים לתוך הטבלאות. הסבר על הגישה לחשבון משתמש שלכם במערכת Postgres מצורפת בסוף התרגיל.

שימו לב! יש לוודא שהקבצים שלכם רצים על מחשבי המעבדה. לא יינתנו נקודות לתשובות שנכשלות בטעינה לתוך מסד הנתונים.

ג) בסעיף זה, תתנסו ביצירת טבלאות, טעינת נתונים ומחיקת טבלאות בעזרת קבצי עזר. **שימו לב: הסעיף הזה להתנסות בלבד. אין תוצר להגשה מסעיף זה.**

הורידו מאתר הקורס את הקבצים : create.sql, drop.sql, ex1.py :

- create.sql מכיל פקודה אשר יוצרת במערכת ה Postgres טבלה אחת בשם enrollment הזהה בצורתה לטבלה המקורית של המידע.
- drop.sql מכיל פקודה המוחקת את הטבלה הנ"ל.
- ex1.py מכיל קוד השולף מתוך קובץ המידע המכווץ (תחת השם enrollment.zip) את שורות המידע, וכותב אותן לתוך קובץ חדש, enrollment.csv ע"י שימוש בפונקציה process_file. שימו לב – קובץ זה רץ באמצעות <u>python3 ומעלה בלבד</u> (במחשבי המעבדה השתמשו בפקודה python3 כדי להריצו).

כעת :

- הריצו את הקוד בקובץ ex1.py וודאו שנוצר לכם הקובץ enrollment.csv.
- התחברו למערכת postgres מתוך התיקיה שבה שמרתם את כל הקבצים על ידי הפקודה : (ההוראות המצורפות בסוף התרגיל, אבל גם רשומות כאן באופן חלקי לנוחיותכם.)

psql -h dbcourse public

- הריצו את הקובץ create.sql ליצירת הטבלה enrollment בעזרת הפקודה

\i create.sql

- התנתקו מהמערכת בעזרת הפקודה

\q

- טענו את הנתונים לתוך הטבלה שיצרתם בעזרת הפקודה

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

אחרי הרצת הפקודה הפלט אמור להיות :

COPY 136596

כלומר 136596 שורות הועתקו לתוך הטבלה.

- התחברו שוב למערכת postgres והריצו את השאילתה הבאה :

SELECT COUNT(*) FROM enrollment;

השאילתה מחזירה את מספר השורות בטבלה enrollment , כך תוודאו שאכן הנתונים נטענו לטבלה כראוי.

- הריצו את הקובץ drop.sql כדי למחוק את הטבלה
\i drop.sql

ד) כעת אתם נדרשים לעדכן את הקבצים create.sql , drop.sql כך שייצרו את הטבלאות המתאימות ליחסים שהגדרתם בסעיף ב. ניתן לשנות מעט את הגדרות הטבלאות על מנת לנצל את תכונות מסד הנתונים (למשל, המסד מאפשר ערכי null). אם בסעיף זה בחרתם ליצור טבלאות שונות מאלו שהגדרתם בסעיף ב, הוסיפו בקובץ ex1.pdf הסבר עבור השינויים שבחרתם לעשות.

שימו לב שבפועל חלק מהמידע שהנחנו שקיים בשלב בניית הדיאגרמה חסר בנתונים, ויש עמודות שמופיעים בהן ערכי null: divisions, latitude, longitude.
בנוסף, גם בעמודה students5_estimated יש שורות בהן המידע חסר, אבל בכל זאת נרצה לשמור את המידע על כך שהיתה הרשמה בשנה מסויימת גם אם לא ידוע מספר הנרשמים.

1. כתבו פקודות create table בתוך הקובץ "create.sql" היוצרות את הטבלאות שלכם.
בפתרון וודאו שכללתם את כל התנאים והמגבלות (key, foreign key, check, etc.) שיכולות להיות מוגדרות על הטבלאות.
2. כתבו פקודות drop table בקובץ "drop.sql" שמוחקות את כל הטבלאות שייצרתם.

התחברו למערכת postgres וודאו שהפקודות שלכם רצות **ללא הודעות שגיאה**.

<u>הערה:</u>

- הקובץ של הנתונים שקיבלתם הוא הקובץ עליו נבחן את התרגילים שלכם.
תבדקו בעצמכם מה האילוצים המתקיימים על הנתונים ע"י ניסוי וטעיה, ואלו האילוצים אותם מצופה מכם לאכוף בקובץ create.sql.
- יש להשתמש בdata-types המתאימים למידע גם אם לא הופיעו במפורש בהרצאה.
ניתן להשתמש בכל ה data-types של PostgreSQL.

ה) בסעיף זה אתם נדרשים לשנות את הקוד בקובץ ex1.py כך שיפצל את המידע לקבצים שונים, בהתאם להגדרות הטבלאות שלכם. זהו למעשה תהליך מאוד סטנדרטי של preprocessing וניקיון של נתונים על מנת לאפשר שאילות או עבודה של data scientist.

בקובץ ex1.py, אתם נדרשים לבצע את השלבים הבאים:

- עבור כל טבלה צרו קובץ עם סיומת csv הנקרא באותו שם כמו הטבלה. יש להקפיד על שם זהה כולל אותיות גדולות וקטנות באנגלית.
- עדכנו את הפונקציה process_file כך שתרשום את המידע הרלוונטי מכל שורה לתוך קבצי ה-csv של הטבלאות השונות. הקפידו לסגור את כל הקבצים שפתחתם בקוד אחרי שאתם מסיימים לכתוב אליהם!
- עדכנו את הפונקציה get_names כך שתחזיר רשימה עם שמות כל הטבלאות שהגדרתם.
השמות צריכים להיות תואמים גם לשמות טבלאות שהגדרתם בסעיף ד, וגם לשמות קבצי ה-csv שהגדרתם בקוד.
שימו לב: יש להחזיר את שמות הטבלאות לפי הסדר הנכון לטעינת נתונים. כלומר, אם יש טבלה A עם אילוץ מפתח זר לטבלה אחרת B, יש להחזיר קודם את B ורק אח"כ את A ברשימה.

כעת, תבדקו שניתן לטעון את הנתונים לכל אחד מהטבלאות בהצלחה. כלומר, תייצרו שוב את הטבלאות. הריצו פקודה של טעינת שורות עבור כל אחד מהטבלאות, לפי אותו סדר שהחזרתם בפונקציה get_names. תוודאו, על ידי שאילתות, שהנתונים נכנסו כראוי. לבסוף תמחקו את הטבלאות.

הערות:

- המידע בקבצי ה-csv שאתם מייצרים צריך להופיע בלי שורות שחוזרות על עצמם כדי שניתן יהיה לטעון את הנתונים באופן תקין לטבלאות מבלי להפר אילוצי מפתח. עבור חלק מהנתונים של האוניברסיטאות, לא כל המידע מוכפל בכל השורות אלא מופיע רק בשורה האחרונה. שימו לב לחלץ את המידע המלא עבור כל אוניברסיטה, גם אם המידע לא מופיע בכל השורות.
למשל, בשורות בקובץ עבור האוניברסיטה העברית, שמופיעות בין שורה 58360 לשורה 58373, אין את כל המידע המעודכן, הוא מופיע במלואו רק בשורה האחרונה מספר 58374.
הכוונה היא שאם יש מידע חסר או סותר בין השורות, יש להתייחס לשורה האחרונה של כל אוניברסיטה כאל המידע הכי עדכני ונכון על האוניברסיטה.
- אם הקוד רץ לכם כמו שצריך אבל בpresubmit אתם רואים שלא נטענו הנתונים- תוודאו שסגרתם בקוד את כל הקבצים שפתחתם בקוד שלכם!
- אם אתם מריצים את הקוד על מחשבי windows לפעמים משום מה מודפסות שורות ריקות לתוך קובץ הCSV. אם זה קורה לכם תנסו לעבוד מרחוק על מחשבי בית הספר וזה אמור לפתור את הבעיה.

- אפשר להניח שהמידע בקובץ המקורי ממויין מיון ראשוני לפי אוניברסיטאות ומיון שניוני לפי שנים.

יש להגיש את הקבצים create.sql, drop.sql, ex1.py בתוך zip ההגשה שלכם.

בהצלחה!

Appendix: Using Postgres

You can access your database account with the command:

psql -h dbcourse public

in the computer labs. After running this command, you can enter queries and DDL commands directly into the command line prompt.
In this exercise it will be more useful for you to write your create and drop table commands in a file, and then this file can be loaded into the database for execution. To do so, use the command

\i a.sql

within the prompt of the database, assuming your commands are in the file "a.sql". Some other useful commands are:

- \q exit psql
- \h [command] help about 'command'
- \d [name] describe table/index/… called 'name'

\dt list tables