

## אוניברסיטת בן-גוריון

### המחלקה להנדסת מערכות תוכנה ומידע / הנדסת תוכנה

#### קורס: בסיסי נתונים – תשפ"א

---

#### מטלה 4 - DB Applications, Triggers, Views

---

שימו לב- אל המטלה מצורף קובץ SQL המכיל פקודות DDL ליצירת בסיס הנתונים. את התרגיל יש לכתוב על בסיס נתונים זה.

**\*\* שימו לב שעל כל השליפות והשדות מסוג DATETIME יש לעבוד עם תאריכים בפורמט YYYY-MM-DD mm:ss:ss לדוגמא: 2019-12-26 09:27:10**

**\*\* שימו לב לא למיין את תוצאות השליפות אם לא התבקשתם לעשות זאת. אנו משווים את תוצאות השליפות שלכם עם התוצאות של השליפות הנכונות. כל שינוי בסדר השדות שנשלפים (ציינו לכם את הסדר) או בסדר הרשומות הנשלפות יגרור שהבדיקות האוטומטיות לא יעברו!**

#### Views

כתבו את ה-Views הבאים ב-SQL :

1. ConstructionEmployeeOverFifty – מציג את כל עובדי הקבלן, בני 50 ומעלה כולל כל השדות של **הישות עובד קבלן** (קודם שדות של עובד ואז של עובד קבלן לפי הסדר שהם שמורים ב-DB).
2. ApartmentNumberInNeighborhood - עבור כל שכונה יש להציג את מספר הדירות השייכות אליה כלומר יש להציג NID ומספר דירות (ApartmentNumber). (קודם NID ואז ApartmentNumber).

#### Triggers

כתבו את הטריגרים הבאים ב-SQL :

1. מחיקת עובד - כאשר מוחקים פרויקט יש גם למחוק את כל העובדים שהיו בפרויקט ולא משויכים לפרויקט אחר כלומר אחרי פעולת מחיקה לא יהיו עובדים אשר לא משויכים לפרויקט.
2. ממשו את האילוף: עובד עירייה לא יכול לנהל יותר מ-2 מחלקות.

#### Mini Application

יש לכתוב את האפליקציה בשפת Python בגרסה 3.6 בלבד, אין להשתמש בספריות מיוחדות מעבר למה שנלמד בתרגול. יש לממש את הפונקציות בתוך הקובץ אשר מגיע עם התרגיל, ניתן לממש פונקציות עזר.

1. עובדי העיריה מאוד אוהבים לעבוד באקסל ומסרבים להשתמש במערכת כדי לנהל את רשימת השכונות. לכן הוחלט להוסיף להם אופציה לטעון את הרשימה מקובץ. כתבו פונקציה load\_neighborhoods אשר מקבלת נתיב לקובץ [csv](#) (קובץ טקסט אשר מופרד בפסיקים ראו קובץ דוגמא מצורף) וטוענת את רשימת השכונות למסד הנתונים כלומר טוענים קובץ בפורמט id, neighborhood. לדוגמא:

1, neighborhood1

2, neighborhood2

2. בעקבות הסכם עבודה של עובדי הקבלן הוחלט לעלות את משכורות כל עובדי הקבלן בני 50 ומעלה כולל ב%3. כתבו פונקציה `update_employee_salaries` אשר מקבלת אחוז (למשל 3) ומעדכנת את משכורת של עובדי הקבלן בני 50 ומעלה.
3. בגלל שינויי תקציב בלתי צפויים יש צורך לחשב את השכר של כל עובדי הקבלן וסה"כ תקציב הפרויקטים.
  - a. כתבו פונקציה המחשבת ומחזירה את סה"כ השכר היומי של כל עובדי הקבלן `get_employee_total_salary`.
  - b. כתבו פונקציה המחשבת ומחזירה את סה"כ השכר של כל תקציב הפרויקטים `get_total_projects_budget`.
4. עקב עליית השכר ראש אגף התקציבים ביקש לנתח את הכנסות העירייה בשנים האחרונות מחניה על מנת לבדוק האם הייתה עליה בהכנסות וניתן להוריד את אחוז הקיצוץ בתקציבי העיריה. כתבו פונקציה `calculateIncomeFromParking` אשר מקבלת שנה (כ-string) ומחזירה את סה"כ ההכנסות מחניה באותה השנה.
5. על מנת למצוא מקורות הכנסה נוספים ראש אגף התקציבים ביקש לקבל מידע על אלו אזורים בעלי ההכנסה הכי גבוהות מחניה וביקש לקבל את הנתונים הבאים: (החזירו מהפונקציות רשימה של `tuples` לדוגמא: `[(3,3),(2,2),(1,1)]`)
  - a. חמשת האזורים הכי רווחיים ממוינים מהרווחי ביותר `get_most_profitable_parking_areas` ומיון משני לפי AID מהגדול לקטן. אם האזורים במקום החמישי והשישי בעלי אותה הכנסה החזירו את האיזור שהID שלו יותר גדול. החזירו לפי הסדר `(ParkingAreaID, Income)`
  - b. כמות מכוניות שונות שחנו בכל אזור ממוינים לפי כמות המכוניות מהגדול לקטן `get_number_of_distinct_cars_by_area`. החזירו לפי הסדר `(ParkingAreaID, Distinct Car Number)`
6. ממשו פונקציית הוספת עובד שהוא לא עובד קבלן ולא עובד רשמי `AddEmployee` אשר מקבלת את כל שדות העובד. דוגמה לקריאה לפונקציה-
7. מטרת התרגיל היא לא לבדוק עקרונות תכנות. הקורס הנ"ל אינו קורס תכנות מתקדם או `sql`. מטרת התרגיל היא לתרגל שלל שימושים בבסיסי נתונים מתוך קוד, ולכן הדגש הוא לא על בדיקות קלט או חיפוש מקרי קצה לדוגמא שמוסיפים עובד לטבלה והמחלקה שלו לא קיימת. זאת לא המטרה של התרגיל!
8. אין להדפיס שום דבר למסך כחלק מהקוד.
9. מתוך סעיפים 6 5b, 4, 3b, 2 יש לפתור רק 6 סעיפים.

## הוראות הגשה

1. עליכם להגיש תיקייה מכווצת אשר נקראת `ID1_ID2` (ת"ז של שני המגישים) אשר מכילה תיקייה הנקראת `Assignment4` שמכילה את הקבצים הבאים-
  - a. קובץ `SQL` עבור `Views & Triggers` אשר נקרא `Triggers & Views.sql`. יש להקפיד להפריד בין הפקודות השונות על ידי ; (נקודה פסיק)
  - b. קובץ `python` אשר נקרא `hw4.py` בו תממשו את הפונקציות ל-`mini application`
  - c. קובץ `B7_DB.db` מכיל נתונים לדוגמא על מנת להקל על תהליך מילוי הDB אך אין להסתמך על זה שהנתונים בו מיצגים את הנתונים אשר ישמשו לבידוק העבודה.
2. העבודה נבדקת אוטומטית, לכן חריגות מההוראות הניתנות בעבודה זו יכולות לגרור הורדת ציון משמעותית, עד כדי ציון 0. אנא הקפידו לעקוב אחר ההוראות! אל תחכו לרגע האחרון בדקו באמצעות המערכת שהעבודה שלכם רצה כנדרש.
3. מומלץ לחלץ את התיקייה שהעלנו לכם למודל, לממש את כל הפונקציות וה-`Triggers & Views`, לכוון מחדש, לקרוא לתיקייה המכווצת לפי ת"ז של שני המגישים ולהגיש למערכת הבדיקות.

4. הגשה **בזוגות בלבד**, אלא אם התקבל אישור מיוחד מסגל הקורס.

5. כתובת מערכת ההגשה- <https://subsys.ise.bgu.ac.il/submission/login.aspx>

יש להגיש לקורס Databases 372 תחת תיקיה Assignment 4.

לצורך מניעת אי הבנות- מצורף הפלט התקין שאמור להתקבל. כל פלט אחר אומר שנזרקה שגיאה בזמן הקימפול שימו לב לכך!

```
- Results

12/26/2019 10:47:46 AM
Submission processing started for file: 4.1221.zip
Found assignment project - DB1
Test files copied to assignment folder
Compilation succeeded
Execution started
Execution done.
Results:
Create ConstructionEmployeeOverFifty View executed successfully
Create ApartmentNumberInNeighborhood View executed successfully
DeleteProject Trigger executed successfully
MaxManger Trigger executed successfully
Mini Application test 1 executed successfully
Mini Application test 2 executed successfully
Mini Application test 3A executed successfully
Mini Application test 3B executed successfully
Mini Application test 4 executed successfully
Mini Application test 5A executed successfully
Mini Application test 5B executed successfully
Mini Application test 6 executed successfully
Total # of errors:
0

*****
```

**בהצלחה!!**