

תרגול מס' 10 - פתרון

מצורפת הצעת פתרון אפשרית לתרגול כיתה שעסק בבניית מסד נתונים ליום הסטודנט וכלל את הטבלאות הבאות:
.Students , Shows , AgudaMembers , Schools

פתרונות לשאלות:

(1) הציגו את כמות הסטודנטים בחלוקה לפי שנות הלימוד שלהם

```
SELECT count(*), year  
FROM students  
GROUP BY year;
```

(2) הציגו את הערים בהם הגיל הממוצע גבוה מ 25

```
SELECT city  
FROM students  
GROUP BY city  
HAVING avg(age) > 25;
```

(3) מצאו את שם הסטודנט המבוגר ביותר

```
SELECT name  
FROM students  
WHERE age = (SELECT max(age) FROM students);
```

4) בוצעה תקלה ובטעות הכניסו את אחד הזמרים שמופיעים לתוך טבלת סטודנטים, על מנת לתקן את הטעות יש לבצע את השלבים הבאים:

א) הוסיפו לטבלת Shows שורה חדשה המכילה את הפרטים הבאים:

a. ID = 70

b. School_ID הינו הערך היושב בשורה של סטודנט מספר 11

c. אין שעת התחלה או שעת סיום

d. שם הזמר הינו השם של הסטודנט המבוגר ביותר

אופציה 1

```
INSERT INTO shows (id, school_id, singer)
```

```
VALUES
```

```
(70,
```

```
(SELECT school_id FROM students WHERE student_ID = 11),
```

```
(SELECT name FROM students WHERE age = (SELECT max(age)  
FROM students))
```

```
);
```

אופציה 2

```
INSERT INTO shows
```

```
VALUES
```

```
(70,
```

```
(SELECT school_id FROM students WHERE student_ID = 11),
```

```
Null,
```

```
Null,
```

```
(SELECT name FROM students WHERE age = (SELECT max(age)  
FROM students))
```

```
);
```

(ב) לאחר הוספת השורה לטבלת Shows דאגו למחוק את פרטי השורה של הזמרת ריטה מטבלת הסטודנטים

```
DELETE FROM students  
WHERE name = 'rita';
```

(5) הציגו את שמות המוסדות שבהן מספר הסטודנטים נמוך מהמוצע

```
SELECT name  
FROM schools  
WHERE numofstudents < ( SELECT avg(numofstudents) FROM schools);
```

(6) בדקו האם הפעלת הפונקציה ROUND (העיגול יתבצע ללא ספרות אחרי הנקודה העשרונית) על הגיל של סטודנט 5 שקולה להפעלת פונקצית CEIL על אותו הערך (השתמשו בפונקציה להשוואת מחרוזות לביצוע בדיקה זו ודאגו להציג את התשובה ללא כפילויות)

```
SELECT DISTINCT STRCMP  
(  
    (SELECT ROUND(age,0) FROM students WHERE student_id = 5),  
    (SELECT CEIL(age) FROM students WHERE student_id = 5)  
);  
FROM students
```

7) הציגו את שמות הזמרים שמופיעים בכל מוסד (דאגו לשמירת מידע דו-כיוונית) דאגו שהתוצאה הסופית תהיה ממוינת במיון יורד לפי שם המוסד

```
SELECT singer
FROM (
    SELECT singer, name
    FROM schools LEFT OUTER JOIN shows
    USING (school_id)
    UNION
    SELECT singer, name
    FROM schools RIGHT OUTER JOIN shows
    USING (school_id)
) as X
ORDER BY name DESC;
```

8) מהם הגילאים של הסטודנטים שגרים באחת מהערים: ירושלים, גבעתיים, רמת-גן

```
SELECT age
FROM students
WHERE city IN ('Ramat-Gan', 'Jerusalem', 'Givatyim');
```

9) הציגו לכל סטודנט את השעות בהן יש לו הופעות במוסד בו הוא לומד (יש צורך בשמירת מידע) רק עבור סטודנטים הגרים בעיר תל-אביב

```
SELECT name, start_hour, end_hour
FROM students st LEFT OUTER JOIN shows sh
ON (st.school_id = sh.school_id)
WHERE city = 'tel-aviv';
```

10) מצאו את מספרי הסטודנטים של החברים באגודה להם יש אישור כניסה עם רכב (אין צורך להציגם), עבור סטודנטים אלו הציגו את שם המוסד בו הם לומדים

```
SELECT name
FROM schools
WHERE school_id = SOME
    (SELECT school_id
     FROM agudamembers JOIN students
     USING (student_id)
     where car_permit = 'V');
```

11) כמה סטודנטים גרים בעיר שבה קיים המוסד עם מספר התלמידים הנמוך ביותר

```
SELECT count(student_id)
FROM students
WHERE city =
    (SELECT city
     FROM schools
     WHERE numofstudents = (SELECT min(numofstudents) FROM schools));
```

12) הציגו את שמות הסטודנטים שגרים בעיר של dotan

```
SELECT name  
FROM students  
WHERE city = (SELECT city FROM students WHERE name='dotan');
```

13) בדקו האם קיימים (שימוש באופרטור EXISTS) סטודנטים באגודה, בהנחה וקיימים הציגו את הגיל הממוצע של כלל הסטודנטים

```
SELECT avg(age)  
FROM students as s  
WHERE EXISTS(SELECT * FROM agudamembers AS a  
WHREE s.student_id = a.student_id);
```