

דו"ח מיני פרויקט בבסיסי נתונים

שרית תיק - 213230048 ואיטי ישראלי - 213712367

אוכל - קייטרינג - שלב ב'

תוכן עניינים

שאלות Select בלי פרמטרים: 2-3

שאלות Update בלי פרמטרים: 4-5

שאלות Delete בלי פרמטרים: 6-7

שאלות Select עם פרמטרים: 8-9

אילוצים: 10

שאלות

שאלות ה-SELECT

שאלת ראשונה:

השאלת מחזירה את השכר הכולל של העובדים בכל קייטרינג כנ"ל קייטרינגים שאין בהם עובדים כלל (יוחזר 0) הרשימה חוזרת בסדר יורד של שכר כולל.

```
SELECT C.Name AS CateringName, COALESCE(SUM(W.Salary), 0) AS TotalSalary
FROM Catering C
LEFT JOIN WorkingIn WI ON C.Catering_ID = WI.Catering_ID
LEFT JOIN Worker W ON WI.Worker_ID = W.Worker_ID
GROUP BY C.Name
ORDER BY TotalSalary DESC;
```

| | CATERINGNAME | TOTALSALARY |
|---|--------------|-------------|
| 1 | Chubby | 68154 |
| 2 | Ritchie | 66464 |
| 3 | Nickel | 54423 |
| 4 | Charlize | 51639 |
| 5 | Brad | 46687 |

שאלת שנייה:

השאלת מחזירה את שני הקייטרינגים הכשרים בעלי משכורת העובדים הכללית הגבוהה ביותר (את השם של הקייטרינג והמשכורת הכללית של העובדים שם)

```
SELECT CateringName, AvgSalary
FROM (
  SELECT C.Name AS CateringName, AVG(W.Salary) AS AvgSalary,
  ROW_NUMBER() OVER (ORDER BY AVG(W.Salary) DESC) AS rn
  FROM Catering C
  JOIN WorkingIn WI ON C.Catering_ID = WI.Catering_ID
  JOIN Worker W ON WI.Worker_ID = W.Worker_ID
  WHERE C.Cosher = 'Y'
  GROUP BY C.Name
)
WHERE rn <= 2;
```

| | CATERINGNAME | AVGSALARY |
|---|--------------|-----------|
| 1 | Angie | 18498 |
| 2 | Penelope | 17637 |

שאלת שלישית:

השאלת מחזירה את המוצרים שנמצאים ביותר מתבשיל אחד (את שם המוצר וכמות התבשילים שהמוצר נמצא)

```
SELECT M.Name AS MaterialName, COUNT(DISTINCT C.Dishes_ID) AS DishCount
FROM Material M
JOIN Contain C ON M.Material_ID = C.Material_ID
GROUP BY M.Name
HAVING COUNT(DISTINCT C.Dishes_ID) > 1
ORDER BY DishCount DESC;
```

| | MATERIALNAME | DISHCOUNT |
|---|--------------|-----------|
| 1 | Coconut | 9 |
| 2 | Rabbit | 8 |
| 3 | Milk | 8 |
| 4 | Commeal | 7 |
| 5 | Chutney | 7 |

שאלתא רביעית:

השאלתא מחזירה את כל הלקוחות שביצעו יותר מהזמנה 1 או לקוחות שביצעו הזמנה שסכומה הכולל עולה על \$500.

```
SELECT Customer_ID, Name, Email, Address, PhoneNumber
FROM Customer
WHERE Customer_ID IN (
  SELECT Customer_ID
  FROM The_Order
  GROUP BY Customer_ID
  HAVING COUNT(Order_ID) > 1
)
UNION
SELECT Customer_ID, Name, Email, Address, PhoneNumber
FROM Customer
WHERE Customer_ID IN (
  SELECT Customer_ID
  FROM The_Order
  GROUP BY Customer_ID
  HAVING SUM(TotalPrice) > 500
);
```

| | CUSTOMER_ID | NAME | EMAIL | ADDRESS | PHONENUMBER |
|---|-------------|--------------|----------------------|----------------------|-------------|
| 1 | 1828 | John Davis | john.dav@example.com | 640 3rd Springfield | 445-1257717 |
| 2 | 2526 | Jane Jones | jane.jon@example.com | 531 Main Riverside M | 227-4001302 |
| 3 | 3131 | John Wilson | john.wil@outlook.com | 408 3rd Bristol NY | 940-4338673 |
| 4 | 3545 | Chris Miller | chris.mi@hotmail.com | 276 Pine Bristol CA | 289-6473149 |
| 5 | 3877 | Katie Miller | katie.mi@example.com | 180 Cedar Riverside | 449-7618322 |

שאלות UPDATE

שאלת ראשונה:

המאגר לפני העדכון

| | DISHS_ID | PRICE | NAME | COSHERTYPE | CATERING_ID |
|---|----------|-------|-----------------|------------|-------------|
| 1 | 2 | 218 | Kerak Telor | Meat | 79 |
| 2 | 4 | 197 | Sate Padang Pad | Meat | 16 |
| 3 | 6 | 66 | Dim Sum | Meat | 769 |
| 4 | 8 | 39 | Fajitas | None | 121 |
| 5 | 10 | 229 | Sate Padang Saw | None | 2 |

שאלת update

השאלת בודקת עבור כל המאכלים שמורכבים ממצרך חלבי אז המאכל הופך להיות חלבי. (חילקתי לשתי תמונות כדי שלא יתפוס לי מדי הרבה מקום בעמוד)

```

UPDATE Dishes D
SET D.CosherType = 'Dairy'
WHERE D.Dishes_ID IN (
    SELECT DISTINCT C.Dishes_ID
    FROM Contain C
    RIGHT JOIN Material M ON C.Material_ID = M.Material_ID
    WHERE M.Type = 'dairy'
);

```

9:1 & Sarititi@XE 93 rows updated in 0 seconds

המאגר לאחר העדכון

| | DISHS_ID | PRICE | NAME | COSHERTYPE | CATERING_ID |
|---|----------|-------|-----------------|------------|-------------|
| 1 | 2 | 218 | Kerak Telor | Meat | 79 |
| 2 | 4 | 197 | Sate Padang Pad | Meat | 16 |
| 3 | 6 | 66 | Dim Sum | Meat | 769 |
| 4 | 8 | 39 | Fajitas | None | 121 |
| 5 | 10 | 229 | Sate Padang Saw | None | 2 |

שאלת שניה:

המאגר לפני העדכון

| | WORKER_ID | WORKHOURES | NAME | SALARY |
|---|-----------|------------|-----------------|--------|
| 1 | 4830 | 9 | Rony Furtado | 11384 |
| 2 | 4840 | 10 | Pierce Hudson | 19315 |
| 3 | 4850 | 6 | Gordie Ammons | 14169 |
| 4 | 4860 | 10 | Natasha Michael | 17568 |
| 5 | 4870 | 12 | Lupe Nicks | 16376 |

שאלת הupdate

השאלת תעדכן את המשכורת של כל העובדים שעובדים בקייטרינג שיש לו לפחות 2 מאכלים שונים ל3,500.

```
UPDATE Worker W
SET W.Salary = 3500
WHERE W.Worker_ID IN (
    SELECT WI.Worker_ID
    FROM WorkingIn WI
    LEFT JOIN Catering C ON WI.Catering_ID = C.Catering_ID
    WHERE C.Catering_ID IN (
        SELECT D.Catering_ID
        FROM Dishes D
        GROUP BY D.Catering_ID
        HAVING COUNT(D.Dishes_ID) >= 2
    )
);
```

14:1 Sarititi@XE 235 rows updated in 0.016 seconds

המאגר לאחר העדכון

| | WORKER_ID | WORKHOURS | NAME | SALARY |
|---|-----------|-----------|-----------------|--------|
| 1 | 4830 | 9 | Rory Furtado | 11384 |
| 2 | 4840 | 10 | Pierce Hudson | 3500 |
| 3 | 4850 | 6 | Gordie Ammons | 14169 |
| 4 | 4860 | 10 | Natasha Michael | 17568 |
| 5 | 4870 | 12 | Lupe Nicks | 16376 |

שאיילת ה DELETE

שאיילת ראשונה:

המאגר לפני המחיקה

| | CUSTOMER_ID | NAME | EMAIL | ADDRESS | PHONENUMBER | PAYMENTDETAILS |
|---|-------------|----------------|----------------------|----------------------|-------------|----------------|
| 1 | 76784 | Chris Miller | chris.mi@hotmail.com | 680 High Riverside N | 628-3754265 | Bit |
| 2 | 53919 | David Williams | david.wi@hotmail.com | 459 Oak Bristol GA | 636-5817095 | CreditCard |
| 3 | 98654 | Laura Brown | laura.brow@yahoo.com | 702 High Franklin IL | 798-2017579 | Cash |
| 4 | 36997 | Alex Williams | alex.wil@hotmail.com | 690 Main Georgetown | 106-3585082 | CreditCard |
| 5 | 56477 | Alex Davis | alex.dav@example.com | 237 Maple Greenfield | 413-9079902 | Bit |
| 6 | 65687 | Chris Jones | chris.jone@yahoo.com | 187 Broadway Bristol | 703-2911076 | Bit |

שאיילת ה delete

השאיילת מוחקת מהמאגר את כל הלקוחות שלא ביצעו שום הזמנה.

```
DELETE FROM Customer C
WHERE C.Customer_ID IN (
  SELECT C.Customer_ID
  FROM Customer C
  LEFT JOIN The_Order O ON C.Customer_ID = O.Customer_ID
  WHERE O.Order_ID IS NULL
);
```

| | | |
|-----|-------------|-----------------------------------|
| 7:3 | Sarititi@XE | 179 rows deleted in 0.016 seconds |
|-----|-------------|-----------------------------------|

המאגר לאחר המחיקה

| | CUSTOMER_ID | NAME | EMAIL | ADDRESS | PHONENUMBER | PAYMENTDETAILS |
|---|-------------|---------------|----------------------|----------------------|-------------|----------------|
| 1 | 76784 | Chris Miller | chris.mi@hotmail.com | 680 High Riverside N | 628-3754265 | Bit |
| 2 | 56477 | Alex Davis | alex.dav@example.com | 237 Maple Greenfield | 413-9079902 | Bit |
| 3 | 65687 | Chris Jones | chris.jone@yahoo.com | 187 Broadway Bristol | 703-2911076 | Bit |
| 4 | 59374 | Laura Brown | laura.br@outlook.com | 462 Main Springfield | 418-2185103 | Bit |
| 5 | 37149 | Michael Davis | michael@outlook.com | 616 2nd Madison FL | 464-7254100 | Bit |

שאיילת ראשונה:

המאגר לפני המחיקה

| | AMOUNT | DISHS_ID | ORDER_ID |
|---|--------|----------|----------|
| 1 | 633690 | 1108 | 69178 |
| 2 | 725135 | 370 | 1118 |
| 3 | 655239 | 42 | 40160 |
| 4 | 471115 | 780 | 86420 |
| 5 | 214624 | 918 | 80904 |

| | ORDER_ID | TOTALPRICE | ORDER_DATE | ADDRESS | PHONENUMBER | CUSTOMER_ID |
|---|----------|------------|------------|---------------------|-------------|-------------|
| 1 | 57077 | 1770673 | 19/09/2020 | 84 Berry Street | 131-9221032 | 10344 |
| 2 | 53411 | 1000000 | 20/07/2020 | 972 France Drive | 175-9749531 | 50747 |
| 3 | 36871 | 1000000 | 28/09/2021 | 24 Zahn Street | 731-3324142 | 66224 |
| 4 | 28283 | 1000000 | 29/08/2023 | 58 Rains Blvd | 995-4963084 | 18219 |
| 5 | 72062 | 9288581 | 20/11/2021 | 18 Hilversum Street | 624-7111650 | 689 |

שאיילת ה delete

השאלתא מוחקת הזמנות שיש בהן תבשילים שעשויים רק מחומרים שהתאריך שלהם פג תוקף.

```
DELETE FROM Belong B
WHERE B.Order_ID IN (
  SELECT O.Order_ID
  FROM The_Order O
  WHERE NOT EXISTS (
    SELECT 1
    FROM Belong BL
    JOIN Contain C ON BL.Dishes_ID = C.Dishes_ID
    JOIN Material M ON C.Material_ID = M.Material_ID
    WHERE BL.Order_ID = O.Order_ID
    AND M.ExpiryDate >= SYSDATE
  )
);

DELETE FROM The_Order O
WHERE O.Order_ID IN (
  SELECT O.Order_ID
  FROM The_Order O
  LEFT JOIN Belong B ON O.Order_ID = B.Order_ID
  WHERE B.Order_ID IS NULL
);
```

21:3 Sarititi@XE 147 rows deleted in 0.015 seconds

21:3 Sarititi@XE 346 rows deleted in 0.031 seconds

המאגר לאחר המחיקה

| | AMOUNT | DISHS_ID | ORDER_ID |
|---|--------|----------|----------|
| 1 | 655239 | 42 | 40160 |
| 2 | 471115 | 780 | 86420 |
| 3 | 214624 | 918 | 80904 |
| 4 | 508938 | 1388 | 54338 |
| 5 | 388520 | 1132 | 29984 |

| | ORDER_ID | TOTALPRICE | ORDER_DATE | ADDRESS | PHONENUMBER | CUSTOMER_ID |
|---|----------|------------|------------|--------------------|-------------|-------------|
| 1 | 28283 | 1000000 | 29/08/2023 | 58 Rains Blvd | 995-4963084 | 18219 |
| 2 | 82852 | 5113651 | 07/03/2021 | 31 Harris | 995-4889461 | 73696 |
| 3 | 66529 | 1000000 | 07/09/2022 | 93rd Street | 762-3802611 | 97712 |
| 4 | 80048 | 774873 | 14/03/2022 | 68 Sylvester Drive | 522-4865122 | 91311 |
| 5 | 2114 | 5271254 | 11/08/2020 | 94 O'Connor Ave | 736-2919258 | 92450 |
| 6 | 81540 | 1000000 | 16/07/2023 | 81 DiFranco Street | 950-1071802 | 58806 |

שאלות עם פרמטרים**שאלת ראשונה:**

השאלת מחזירה את כל המוצרים שהתאריך תפוגה שלהם הוא בין התאריכים שהזין המשתמש.

```
SELECT M.Name AS MaterialName, M.Type, M.ExpiryDate, COUNT(DISTINCT C.Dish_ID) AS DishCount
FROM Material M
LEFT JOIN Contain C ON M.Material_ID = C.Material_ID
WHERE M.ExpiryDate BETWEEN TO_DATE(:start_date, 'DD/MM/YYYY') AND TO_DATE(:end_date, 'DD/MM/YYYY')
GROUP BY M.Name, M.Type, M.ExpiryDate
HAVING COUNT(DISTINCT C.Dish_ID) > 0
ORDER BY M.ExpiryDate DESC;
```

| | MATERIALNAME | TYPE | EXPIRYDATE | DISHCOUNT |
|---|-----------------|------------|------------|-----------|
| 1 | Tapoca | dried food | 31/07/2025 | 1 |
| 2 | Tomato Paste | meat | 29/07/2025 | 1 |
| 3 | Salva | dairy | 28/07/2025 | 1 |
| 4 | Beans | dairy | 27/07/2025 | 3 |
| 5 | Red Wine Vinega | dried food | 27/07/2025 | 2 |
| 6 | limac | meat | 26/07/2025 | 2 |

שאלת שנייה:

השאלת מחזירה את כל הקייטריגים הכשרים שכמות העובדים שלהם היא 3 ומעלה, התוצאות החוזרות יהיו ממויינות לפי כמות העובדים בסדר יורד.

```
SELECT C.Name AS CateringName, C.Cosher, COUNT(WI.Worker_ID) AS TotalWorkers
FROM Catering C
LEFT JOIN WorkingIn WI ON C.Catering_ID = WI.Catering_ID
WHERE C.Cosher = :<name="cosher" list="Y,N" type="string">
GROUP BY C.Name, C.Cosher
HAVING COUNT(WI.Worker_ID) >= 3
ORDER BY TotalWorkers DESC;
```

| | CATERINGNAME | COSHER | TOTALWORKERS |
|---|--------------|--------|--------------|
| 1 | Rebeka | Y | 6 |
| 2 | Charlie | Y | 5 |
| 3 | Joaquin | Y | 5 |
| 4 | Derrick | Y | 5 |
| 5 | Ritchie | Y | 4 |
| 6 | Rebecca | Y | 4 |

שאלת שלישית:

השאלת מחזירה כמה פרמטרים (חדשים שחושבו וישנים) של כל ההזמנות שה-ID שלהם נבחר על ידי המשתמש.

```
SELECT
O.Order_ID,
O.TotalPrice,
O.Order_Date,
COUNT(B.Dish_ID) AS DishCount,
AVG(O.TotalPrice) OVER () AS AvgOrderPrice,
(SELECT COUNT(*) FROM The_Order WHERE Customer_ID = O.Customer_ID) AS TotalOrdersByCustomer
FROM The_Order O
LEFT JOIN
Belong B ON O.Order_ID = B.Order_ID
WHERE
O.Order_ID IN (:<name="order_ID" list="SELECT Order_ID FROM The_Order" multiselect="yes">)
GROUP BY
O.Order_ID, O.TotalPrice, O.Order_Date, O.Customer_ID
ORDER BY
O.Order_Date DESC;
```

| | ORDER_ID | TOTALPRICE | ORDER_DATE | DISHCOUNT | AVGORDERPRICE | TOTALORDERSBYCUSTOMER |
|---|----------|------------|------------|-----------|---------------|-----------------------|
| 1 | 6819 | 3092466 | 16/06/2023 | 2 | 2046233 | 1 |
| 2 | 13698 | 1000000 | 06/02/2022 | 4 | 2046233 | 1 |

שאלת רביעית:

השאלתא מחזירה את כל התבשילים שכוללים בתוכם את המרכיבים שהשתמש בחר.

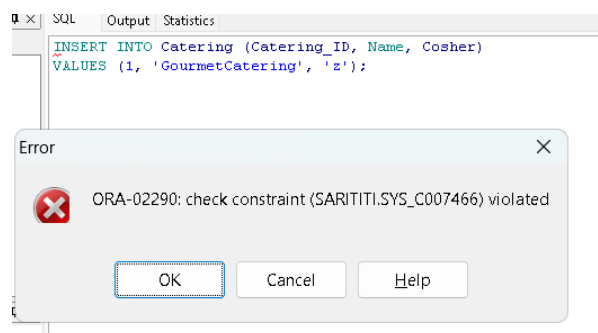
```
SELECT DISTINCT D.Dish_ID, D.Name AS DishName, D.Price, D.CosherType, AVG(D.Price) OVER () AS AvgDishPrice
FROM Dish D
LEFT JOIN Contain C ON D.Dish_ID = C.Dish_ID
WHERE C.Material_ID IN (cname="Materials" list="SELECT Material_ID FROM Material" multiselect="yes" >)
GROUP BY D.Dish_ID, D.Name, D.Price, D.CosherType
ORDER BY D.Price DESC;
```

| | DISHS_ID | DISHNAME | PRICE | COSHERTYPE | AVGDISHPRICE |
|---|----------|------------|-------|------------|--------------|
| 1 | 874 | Sate Jamur | 145 | None | 73 |
| 2 | 362 | Krokot | 72 | None | 73 |
| 3 | 818 | Tongteng | 55 | Dairy | 73 |
| 4 | 204 | Lakis | 20 | Dairy | 73 |

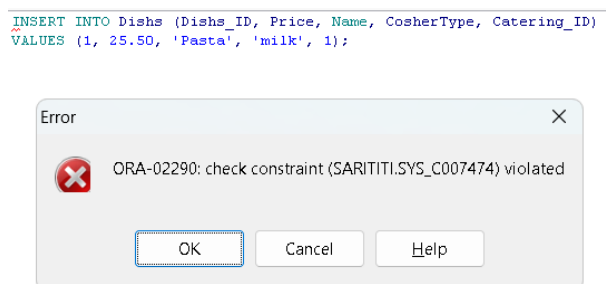
אילוצים

את אילוץ ה-CHECK ביצענו עוד בשלב הראשון, שני אילוצי CHECK שאותם כתבנו בדו"ח של השלב הראשון.

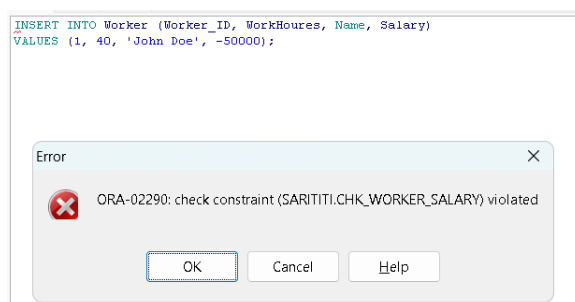
לא נתן להכניס לערך Coshers ערכים השונים מ 'N'/'Y' המסמלים כן או לא ולכן יש הפרה לאילוצן.



לא נתן להכניס לערך Coshers ערכים השונים מ 'MEAT'/'DAIRY'/'NONE' המסמלים את כשרות המאכל, חלבי, בשרי או פרווה ולכן יש הפרה לאילוצן.



הוספנו אילוצן נוסף על טבלת worker שהמשכורת לא תוכל להיות 0 ומטה ולכן גם כאן יש הפרה של האילוצן.



נוסיף עוד שני אילוצי DEFAULT:

```
ALTER TABLE Catering
MODIFY Coshers DEFAULT 'N';

ALTER TABLE Material
MODIFY Type DEFAULT 'Unknown';
```

(סוף)