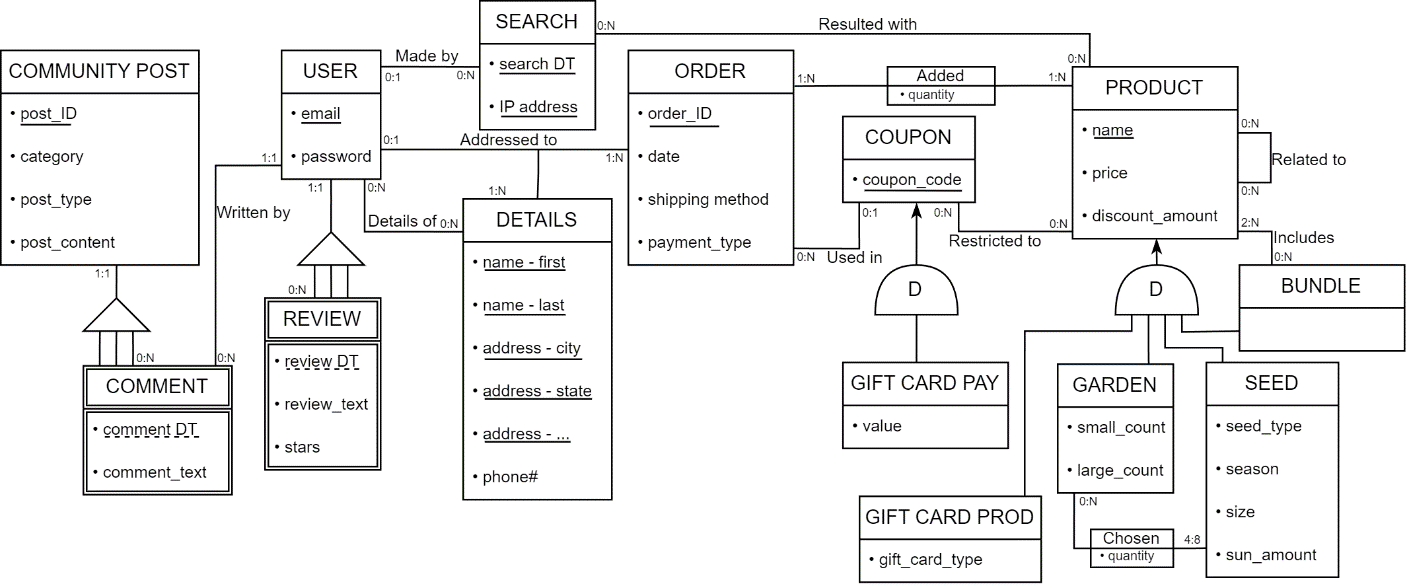
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| מס' קבוצה | אתר | | | תאריך הגשה |
| 16 | [seedsheets.com](http://seedsheets.com) | | | 19/02/2024 |
| חברי הצוות – מספרי ת.ז | | | | |
| 318334901 | | 211680509 | 318255882 | |

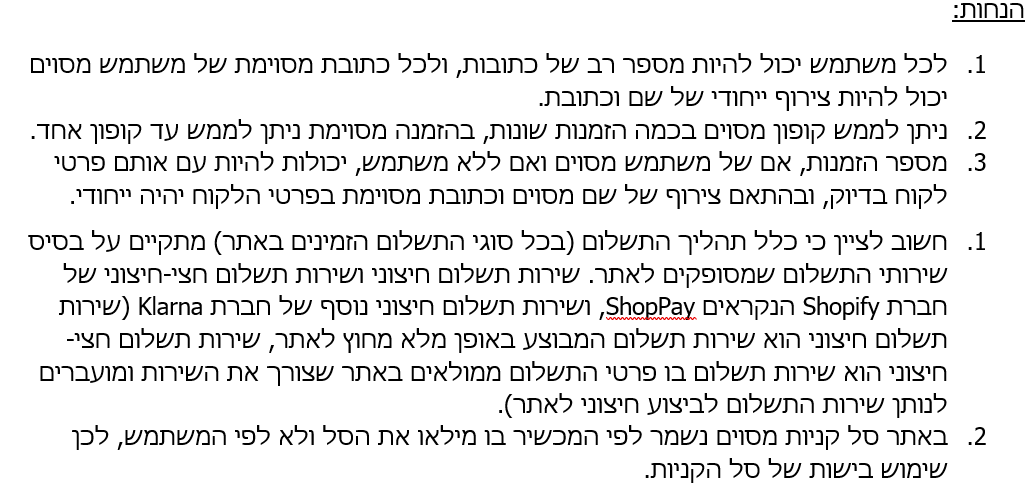
## פרויקט בסיסי נתונים

## מטלה מקדימה – ה-ERD שהוגש בחלק א'

**התרשים בהגשה הקודמת**



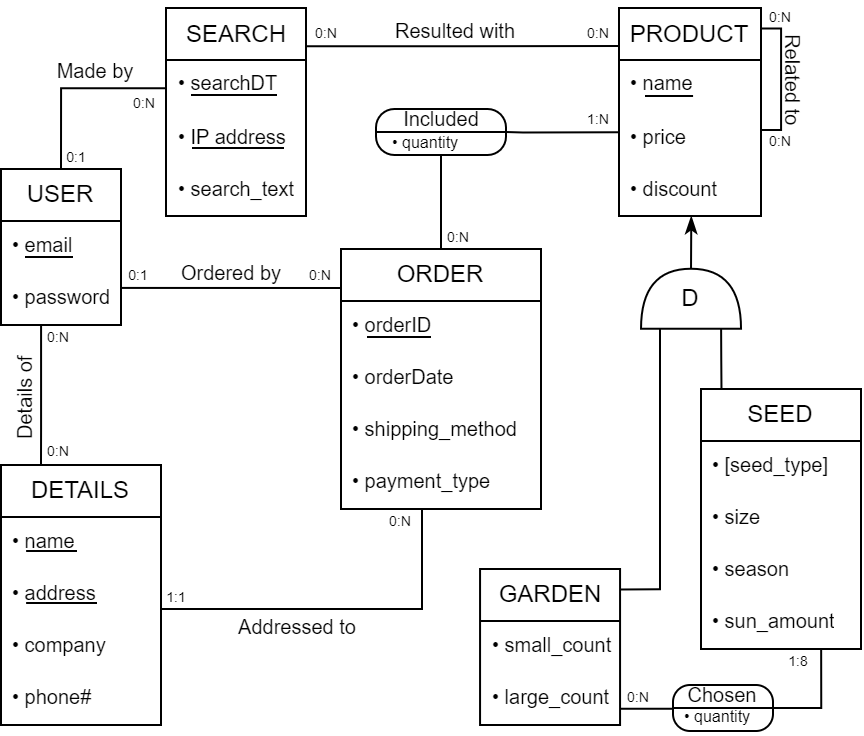
**ההנחות שנרשמו בהגשה הקודמת**



**ההערות שניתנו להגשה הקודמת**

1. התבקשתם ליצור מודל פשוט ככל הניתן המכיל בין 5-8 ישויות ומתייחס לתהליך העסקי המרכזי שלכם. יש לכם 14 ישויות. ישויות ללא הצדקה.
2. סימונים לא נכונים של קשרים- תכונה של קשר יסומן בתאך צורה אובלית ולא בתוך מלבן. מלבן מייצג ישות בלבד.
3. איך מתחשבת ההנחה? היא קבועה? אם לא, זה שדה מחושב והוא לא ייכנס לERD
4. איך נקבע מחיר של גינה מעוצבת? האם היא קבועה או תלויה בעיצוב?  
   אם תלויה בעיצוב, שדה זה יהיה מחושב ולא נכניס אותו לERD. נסו להבין מה באמת קובע את המחיר.
5. מה משמעות הקשר הרקורסיבי?
6. האם באתר אפשר לקנות רק SEED? יש לו מחיר משל עצמו? הוא ממש מוצר בפני עצמו או שהוא רק חלק מגינה מעוצבת?
7. מה המשמעות של השדות של גינה? איזה ערכים הם מקבלים? התכוונתם לכמות הplants? יש להשתמש בשמות אינפורמטיבים. בנוסף, אם באמת התכוונתם לplants שימו לב שאין הרבה אופציות והקומבינתיות בין הקטן לגדול קבועות. במקרה כזה הייתי משתמשת במקום שדות אלו בtype על פי שמות האופציות באתר
8. לא יכול להיות מוצר שלא הופיע באף הזמנה? קרדינליות 0:N יותר הגיונית פה
9. במודל הרשה שיש רק בן אחד אין משמעות לייצוג D או O, נשאיר ריק
10. האם קופונים וגיפט קארדים קלוונטי לתהליך העסקי שלכם?
11. כשנשמור חיפושים נרצה לדעת את תוכן החיפוש.
12. משתמש חייב לעשות הזמנה? קרדינליות 0:N יותר הגיוני פה.
13. מה מטרת הישות הזאת? האם כדי לבצע חיפוש באתר אני צריכה להכניס פרטי כתובת?  
    למה שלאותו משתמש יהיו כמה שמות?  
    מה זה כתובת שלוש נקודות?  
    ולמה אין קשר להזמנה? איפה הכתובת באה לידי ביטוי בהזמנה?
14. על מה הלקוח נותן ביקורת? על מוצא כלשהו או על האתר?  
    אם על מוצר, חסר קשר למוצר ואם על האתר זה לא קשור לתהליך העסקי ולכן לא רלוונטי
15. גם כאן, על מה נכתבת תגובה? האם יש לזה קשר לתהליך העסקי שלכם?

## מטלה 1 (40%) – עיצוב קונספטואלי (תרשים ישויות-קשרים)



## הנחות מודל ה- ERD

|  |  |  |
| --- | --- | --- |
| **מרכיב** | **הנחות** | **הערות נוספות** |
| Entity: USER   * email * password | * מפתח ראשי: email – לכל USER יש כתובת email אחת ויחידה. |  |
| Entity: SEARCH   * searchDT * IP\_address * Search\_text | * מפתח ראשי: צירוף השדות searchDT, IP\_address – כל חיפוש מסוים יכול להיעשות בתאריך וזמן מסוים במכשיר בעל כתובת IP מסוימת. * הקשר "Made by" – חיפוש יכול להיעשות על ידי משתמש רשום מסוים או משתמש מזדמן מסוים, בנוסף, כל משתמש (רשום או לא) יכול לעשות אינספור חיפושים או לא לעשות חיפושים כלל. לכן הקרדינליות של קשר זה בצד הישות SERACH היא 0:N ובצד הישות USER היא 0:1. |  |
| Entity: DETAILS   * name * address * company * phone# | * name – שדה זה מכיל את השם המלא של משתמש רשום/לא רשום. באתר שם הלקוח מיוצג כשתי שדות, אך היות ושתי השדות הללו מייצגות תוכן מאותו אופי ובעלות משמעות דומה נתייחס אליהם כאחד. * Address – שדה זה מכיל את הכתובת המלאה של משתמש רשום/לא רשום. באתר כתובת הלקוח מיוצג כחמישה שדות, אך היות וחמשת השדות הללו מייצגות תוכן מאותו אופי ובעלות משמעות דומה נתייחס אליהם כאחד. * מפתח ראשי: צירוף השדות name, address – כל צירוף הינו ייחודי באתר. לכל צירוף כזה צריך להיות שם חברה יחיד (company) ומספר טלפון יחיד (phone#). |  |
| הקשר "Details of" בין USER לDETAILS | * פרטי לקוח מסוימים יכולים להיות שמורים באופן זהה אצל אינספור משתמשים רשומים שונים או לאף משתמש רשום. לכן הקרדינליות של קשר זה בצד הישות USER היא 0:N. * למשתמש רשום מסוים יכולים להיות שמורים אינספור פרטי לקוח שונים או בכלל לא. לכן הקרדינליות של קשר זה בצד הישות DETAILS היא 0:N. |  |
| Entity: ORDER   * orderID * orderDate * shipping\_method * payment\_type | * מפתח ראשי: orderID – מספר סידורי ייחודי, מזהה של הזמנה מסוימת. * payment\_type – שדה המכיל את שם שירות התשלום החיצוני שבו השתמש הלקוח בעת התשלום על ההזמנה. * הקשר "Ordered by" – הזמנה מסוימת יכולה להיעשות ע"י משתמש רשום/לא רשום אחד. לכן הקרדינליות של קשר זה בצד הישות USER היא 0:1. משתמש רשום מסוים יכול לעשות אינספור הזמנות או בכלל לא. לכן הקרדינליות של קשר זה בצד הישות ORDER היא 0:N * הקשר "Addressed to" – בהזמנה מסוימת יכולים להירשם סט פרטי לקוח אחד בלבד. לכן הקרדינליות של קשר זה בצד הישות DETAILS היא 1:1. פרטי לקוח מסוימים יכולים להירשם באינספור הזמנות או בכלל לא. לכן הקרדינליות של קשר זה בצד הישות ORDER היא 0:N. | כפי שצוין בחלק א', כלל תהליך התשלום, על פני כל סוגי התשלום הזמינים באתר, מתקיים על בסיס שירותי תשלום חיצוניים אותם האתר צורך. שניים משירותי התשלום נצרכים מחברת Shopify ונקראים ShopPay, אחד שירות חצי-חיצוני והשני שירות חיצוני, והשלישי נצרך מחברת Klarna, שירות חיצוני. לכן במידת הצורך כדי להשיג את פרטי התשלום במועד מאוחר יותר נשמר שם שירות התשלום ממנו צריך לייבא את פרטי התשלום |
| Entity: PRODUCT   * name * price * discount | * מפתח ראשי: name – לכל מוצר קיים שם ייחודי * price – מחירו של מוצר מסוים יהיה קבוע וניתן לשינוי ע"י בעל העסק בלבד, לא שדה מחושב. שדה זה יכיל מספרים ממשיים גדולים מ0. * discount – הנחה על מוצר מסוים אשר הערך הדיפולטי שלה הוא 0, ערכה של ההנחה תוכל להיות קטנה או שווה לערכו של מחיר המוצר. שדה זה יכיל מספרים ממשיים גדולים או שווים ל0. * קיימים סוגים שונים של מוצרים בעלי תכונות שונות אחת מהשנייה וקיימים המוצרים בעלי התכונות הבסיסיות של מוצר. לכן, ישות PRODUCT היא ישות אב כחלק ממודל הורשה בדידה וחלקית. |  |
| הקשר "Resulted with" בין PRODUCT וSEARCH | * כל חיפוש מסוים יכול להוביל להצגת אינספור מוצרים או את אף אחד מהמוצרים. לכן הקרדינליות של קשר זה בצד הישות PRODUCT היא 0:N. * כל מוצר מסוים יכול להיות מוצג כתוצאה של אינספור חיפושים או של שום חיפוש. לכן הקרדינליות של קשר זה בצד הישות SEARCH היא 0:N. |  |
| הקשר הרקורסיבי "Related to" בין PRODUCT לעצמו | * כל מוצר מסוים יכול להיות מוצר נלווה לאינספור מוצרים או לאף אחד מהמוצרים, ולכל מוצר מסוים יכולים להיות אינספור מוצרים נלווים או אף לא אחד. לכן הקרדינליות של הקשר הרקורסיבי בשתי צדדי הקשר תהיה 0:N. |  |
| הקשר "Included" בין PRODUCT וORDER   * quantity | * כל מוצר מסוים יכול להיכלל באינספור הזמנות או באף אחת מההזמנות. לכן הקרדינליות של קשר זה בצד הישות ORDER היא 0:N. * בכל הזמנה מסוימת יכולים להיכלל אינספור מוצרים ולכל הפחות מוצר אחד. לכן הקרדינליות של קשר זה בצד הישות PRODUCT היא 1:N. * quantity – כמות של מוצר מסוים שנכלל בהזמנה מסוימת. שדה זה יכיל מספרים שלמים גדולים מ0. |  |
| Child Entity: SEED   * ]seed\_type[ * size * season * sun\_amount | * ישות בת במודל ההורשה של ישות מוצר. * זרעים מסוג מסוים יכולים להימכר באופן עצמאי ללא הכללתם בגינה מסוימת או בעסקת חבילה מסוימת. * [seed\_type] – קטגוריה שמתארת את סוג הצמח, זרע מסוים יכול להימצא במספר קטגוריות. * size – גודל החלקה שהזרע לוקח. יכול להיות או גדול או קטן. * season, sun\_amount – תכונות המכילות מידע אינפורמטיבי על הזרע המוצגות ללקוח באתר. |  |
| Child Entity: GARDEN   * small\_count * large\_count | * ישות בת במודל ההורשה של ישות מוצר. * באתר קיימות גינות בעיצוב אישי וגינות מוכנות מראש אשר. * small\_count, large\_count – תכונות המכילות מספרים ערך שלם לא שלילי. צירופי הערכים האפשריים בשדות אלו יהיו אחד מהשלושה הבאים: small\_count = 8 וגם large\_count = 0 small\_count = 5 וגם large\_count = 1 small\_count = 2 וגם large\_count = 2 |  |
| הקשר "Chosen" בין SEED ו GARDEN   * quantity | * כל זרע מסוים יכול להיכלל באינספור גינות או באף אחת מהן. לכן הקרדינליות של קשר זה בצד הישות GARDEN היא 0:N. * בכל גינה מסוימת יכולים להיכלל בין 1 ל8 זרעים. לכן הקרדינליות של קשר זה בצד הישות SEED היא 1:N. * quantity – כמות זרעים מסוג מסוים בגינה.  שדה זה יכיל מספרים שלמים גדולים מ0. * בגינה מסוימת יכללו בדיוק כמות הזרעים שהיא יכולה להכיל, לפי הכמויות וסוגי החלקות המצוינים בישות הגינה המסוימת. |  |

## מטלה 2 (30%) – עיצוב לוגי (מודל נתונים טבלאי)

* **PRODUCTS** ( **Name**, Price, Discount )
* **USERS** ( **Email**, Password)
* **SEARCHES** ( **SearchDT**, **IP\_address**, Email ( USERS ), Search\_text )
* **RELATIONS** ( **Product1** ( PRODUCTS ), **Product2** ( PRODUCTS ) )
* **RESULTS** ( **Name** ( PRODUCTS ), { **SearchDT**, **IP\_address** } ( SEARCHES ) )
* **GARDENS** ( **Name** ( PRODUCTS ), Small\_count, Large\_count )
* **SEEDS** ( **Name** ( PRODUCTS ), Size, Season, Sun\_amount )
* **SEED\_TYPES** ( **Name** ( SEEDS ), **Type** )
* **CHOSENS** ( **Garden** ( GARDENS ), **Seed** ( SEEDS ), Quantity )
* **DETAILS** ( **Name**, **Address**, Company, Phone# )
* **DETAILS\_OF** ( **Email** ( USERS ), { **Name**, **Address** } ( DETAILS ) )
* **ORDERS** ( **OrderID**, Email ( USERS ), { Name, Address } ( DETAILS ), OrderID,  
   Shipping\_method, Payment\_type )
* **INCLUSIONS** ( **OrderID** ( ORDERS ), **Name** ( PRODUCTS ), Quantity )

**הנחות והסברים נוספים:**

* **ביטוי הקשר Ordered by שבין USER ל-ORDER** – קשר יחיד לרבים. על פי חוקי המעבר למודל הטבלאי כאן בהכרח הקשר יתבטא כשדה נוסף בצד הרבים של הקשר (ORDER) שמגלם מפתח זר לצד היחיד של הקשר (USER).  
  לציין כי האתר מאפשר לבצע הזמנה גם ללא משתמש רשום. הדבר בא לידי ביטוי במפתח הזר של USER בטבלת ORDERS, בכך שאם מבצע ההזמנה אינו משתמש רשום, הערך בשדה זה יהיה ריק.
* **ביטוי הקשר Made by שבין USER ל-SEARCH** – קשר יחיד לרבים. גם כאן באופן זהה ומאותם סיבות הקשר יתבטא כשדה נוסף בצד הרבים של הקשר (SEARCH) שמגלם מפתח זר לצד היחיד של הקשר (USER).
* **ביטוי הקשר Addressed to שבין ORDER ל-DETAILS** – קשר יחיד לרבים. בעקבות חוקי המעבר למודל הטבלאי, הקשר יתבטא כזוג שדות נוספים בצד הרבים של הקשר (ORDER) שיחד יהוו מפתח זר לצד היחיד של הקשר (DETAILS).  
  חשוב לציין ששדה זה בהכרח לא יכיל ערך ריק בעקבות קרדינליות הקשר בצד היחיד.
* **ביטוי הקשר Details of שבין USER ל-DETAILS** – קשר רבים לרבים. על פי חוקי המעבר למודל הטבלאי כאן נדרש ליצור טבלה נפרדת עם שתי מפתחות זרים, אחד שמגלם בשדה יחיד את המפתח הראשי של USER והשני שמגלם בשתי שדות את המפתח הראשי של DETAILS.  
  שתי המפתחות הזרים הללו יחד יהוו מפתח ראשי לטבלה החדשה, בטבלה זו לא יהיו שדות נוספים.
* **ביטוי הקשר Resulted with שבין SEARCH ל-PRODUCT** – קשר רבים לרבים. באופן זההלקשר הקודם, ניצור טבלה נפרדת בה שתי מפתחות זרים, הראשון בעל שדה אחד שמגלם את המפתח הראשי של PRODUCT, והשני שמגלם באמצעות שני שדות את המפתח הראשי של SEARCH.  
  שני המפתחות הזרים הללו יחד יהוו מפתח ראשי לטבלה החדשה ולא יהיו שדות נוספים בטבלה זו.
* **ביטוי הקשר הרקורסיבי Related to של PRODUCT** – קשר רקורסיבי רבים לרבים. בדומה לקשרים הקודמים, ניצור טבלה חדשה בה יהיו הפעם רק שני שדות, כל אחד יהווה מפתח זר נפרד וביחד יהוו מפתח ראשי לטבלה חדשה. שני המפתחות יצביעו על שני מופעים שונים של ישות מוצר.
* **ביטוי הקשר Included שבין ORDER ל-PRODUCT** – קשר רבים לרבים. בדומה לקשרים הקודמים, ניצור טבלה חדשה כדי לייצג את הקשר. בנוסף למפתחות הזרים אשר ביחד יהוו מפתח ראשי, יהיה שדה עבור התכונה quantity המצוינת במודל הERD-.  
  מפתח זר ראשון יגלם את המפתח הראשי של PRODUCT, מפתח זר שני יגלם את המפתח הראשי של ORDER.
* **ביטוי הקשר Chosen שבין GARDEN ל-SEED** – קשר רבים לרבים. הטבלה תכיל שלושה שדות: הראשון מפתח זר לGARDEN, השני מפתח זר לSEED כך ששניהם יחד יהיו המפתח הראשי, ושדה שלישי עבור התכונה quantity המצוינת במודל ה-ERD.
* **ביטוי מודל ההורשה שבין PRODUCT ל-SEED ול-GARDEN** – מודל הורשה בדיד וחלקי. על פי חוקי המעבר למודל הטבלאי קיימת הבחירה בין טבלה אחת לכלל הישויות או טבלה לכל אחת מהישויות במודל ההורשה, במקרה שלנו באמצעות שלוש טבלאות.  
  הדרך שנבחרה לביטוי מודל ההורשה הינה באמצעות טבלה לכל ישות, כדי להימנע מערכי NULL רבים שבהכרח יהיו במימוש באמצעות טבלה אחת. זאת מכיוון שערכי ה-NULL הרבים יתפסו מקום בזכירון ויעמדו בדרכנו בעת תחקור מסד הנתונים.

# פירוט שדות בסיס הנתונים והערכת גודלם

| **Table** | **Field** | | **Data Type** | **Size (Bytes)** |
| --- | --- | --- | --- | --- |
| **PRODUCTS** | Name | | Varchar(80) | 80 |
| Price | | Smallmoney | 4 |
| Discount | | Smallmoney | 4 |
| **Total Record Size (Bytes)** | | | **88** |
| **USERS** | Email | | Varchar(40) | 40 |
| Password | | Varchar(30) | 30 |
| **Total Record Size (Bytes)** | | | **70** |
| **SEARCHES** | SearchDT | | Datetime | 8 |
| IP\_address | | Varchar(15) | 15 |
| Email **(USERS)** | | Varchar(40) | 40 |
| Search\_text | | Varchar(80) | 80 |
| **Total Record Size (Bytes)** | | | **143** |
| **RESULTS** | Name **(PRODUCTS)** | | Varchar(80) | 80 |
| SearchDT | **(SEARCHES)** | Datetime | 8 |
| IP\_address | Varchar(15) | 15 |
| **Total Record Size (Bytes)** | | | **103** |
| **SEEDS** | Name **(PRODUCTS)** | | Varchar(80) | 80 |
| Size | | Char(5) | 5 |
| Season | | Varchar(6) | 6 |
| Sun\_amount | | Varchar(30) | 30 |
| **Total Record Size (Bytes)** | | | **121** |
| **SEED\_TYPES** | Name **(SEEDS)** | | Varchar(80) | 80 |
| Type | | Varchar(20) | 20 |
| **Total Record Size (Bytes)** | | | **100** |
| **GARDENS** | Name **(PRODUCTS)** | | Varchar(80) | 80 |
| Small\_count | | Tinyint | 1 |
| Large\_count | | Tinyint | 1 |
| **Total Record Size (Bytes)** | | | **82** |
| **CHOSENS** | Garden **(GARDENS)** | | Varchar(80) | 80 |
| Seed **(SEEDS)** | | Varchar(80) | 80 |
| Quantity | | Tinyint | 1 |
| **Total Record Size (Bytes)** | | | **161** |
| **DETAILS** | Name | | Varchar(40) | 40 |
| Address | | Varchar(150) | 150 |
| Company | | Varchar(40) | 40 |
| Phone# | | Varchar(30) | 30 |
| **Total Record Size (Bytes)** | | | **260** |
| **DETAILS\_OF** | Email **(USERS)** | | Varchar(40) | 40 |
| Name | **(DETAILS)** | Varchar(40) | 40 |
| Address | Varchar(150) | 150 |
| **Total Record Size (Bytes)** | | | **230** |
| **ORDERS** | OrderID | | Int | 4 |
| Email **(USERS)** | | Varchar(40) | 40 |
| Name | **(DETAILS)** | Varchar(40) | 40 |
| Address | Varchar(150) | 150 |
| OrderDate | | Date | 3 |
| Shipping\_method | | Varchar(100) | 100 |
| Payment\_type | | Varchar(7) | 7 |
| **Total Record Size (Bytes)** | | | **344** |
| **RELATIONS** | Product1 **(PRODUCTS)** | | Varchar(80) | 80 |
| Product2 **(PRODUCTS)** | | Varchar(80) | 80 |
| **Total Record Size (Bytes)** | | | **160** |
| **INCLUSIONS** | OrderID **(ORDERS)** | | Int | 4 |
| Name **(PRODUCTS)** | | Varchar(80) | 80 |
| Quantity | | Tinyint | 1 |
| **Total Record Size (Bytes)** | | | **85** |

**הנחות והסברים נוספים:**

* בשם של מוצר מסוים יהיו לכל היותר 80 תווים.
* לכתובת Email מסוימת, לשם של לקוח מסוים ולשם של חברה מסוימת יהיו באורך של לכל היותר 40 תווים.
* בשדה Type יהיו ערכים הקיימים ברשימה מוגדרת מראש שיכולים להיות בשדה זה.
* בשדה המתאר כתובת יהיו ערכים בהם יש לכל היותר 150 תווים, זאת משום שבכל ערך יצוינו כלל הפרטים של הכתובת ברצף, מדינה, מחוז, עיר, רחוב, מס' בית ומיקוד.
* בשדה Size יוכלו להיות הערכים "Small" או "Large" בלבד.
* בשדה Season יוכלו להיות הערכים "Summer", "Spring", "Winter" או "Fall" בלבד.
* בשדה Payment\_type יוכלו להיות הערכים "Klarna" או "ShopPay" בלבד, זאת כך ש"Klarna" מייצגת את שיטת התשלום דרך שירות התשלום החיצוני של Klarna שהאתר צורך, וש"ShopPay" מייצגת את שתי שיטות התשלום דרך שירותי התשלום החיצוני והחצי-חיצוני של Shopify.
* מחיר של מוצר מסוים, ובהתאם גם הנחה, יהיה לכל היותר 214,748.3647.
* כל השדות שמהווים כמות כלשהי (Quantity) יוכלו להכיל ערך של 255 לכל היותר.
* בשדה Phone# יוכלו להיות ערכים מהפורמט של 10 ספרות ברצף בלבד.
* בשדה IP\_address יוכלו להיות ערכים מהפורמט x.x.x.x כאשר כל x יכול להיות מספר שלם בתחום שבין 0 ל255.
* יהיו 2,247,483,647 הזמנות לכל היותר.
* **שימו לב:** הנחה זו מתייחסת באופן ספציפי לעיצוב אישי של גינה ומביעה הסבר לאיך ולמה לא ייווצרו כפילויות בטבלה GARDENS.  
  כאשר לקוח מזמין גינה בעיצוב אישי מסוים, אם הגינה בעיצוב הזה הוזמנה בעבר, תשמר רשומה להזמנת הלקוח המתארת קשר אל הגינה בעלת עיצוב זה הקיימת במאגר הנתונים.  
  אם הגינה בעיצוב הזה לא הוזמנה בעבר, הגינה תשמר כמוצר חדש עם בשם ייעודי המתאים לעיצוב החדש, זאת על פי תבנית מוגדרת מראש בבסיס הנתונים של האתר, בהתאם יישמרו רשומות נוספות המתארות את הקשרים הרלוונטיים לעיצוב החדש, לבסוף תשמר רשומה להזמנת הלקוח המתארת קשר אל הגינה בעיצוב החדש כפי שנשמרה כמוצר חדש.

## מטלה 3 (30%) - עיצוב פיסי (בסיס נתונים פעיל על שרת RDBMS)

# SCRIPT לבניית הטבלאות

CREATE TABLE dbo.PRODUCTS (

Name Varchar(80) NOT NULL,

Price Smallmoney NOT NULL,

Discount Smallmoney NOT NULL DEFAULT 0,

CONSTRAINT Pk\_product PRIMARY KEY (Name))

CREATE TABLE dbo.RELATIONS (

Product1 Varchar(80) NOT NULL,

Product2 Varchar(80) NOT NULL,

CONSTRAINT Pk\_relation PRIMARY KEY (Product1, Product2),

CONSTRAINT Fk\_Product1 FOREIGN KEY (Product1)

REFERENCES dbo.PRODUCTS (Name),

CONSTRAINT Fk\_Product2 FOREIGN KEY (Product2)

REFERENCES dbo.PRODUCTS (Name))

CREATE TABLE dbo.USERS (

Email Varchar(40) NOT NULL,

Password Varchar(30) NOT NULL,

CONSTRAINT Pk\_user PRIMARY KEY (Email))

CREATE TABLE dbo.SEARCHES (

SearchDT Datetime NOT NULL,

IP\_address Varchar(15) NOT NULL,

Email Varchar(40) NULL,

Search\_text Varchar(80) NOT NULL,

CONSTRAINT Pk\_search PRIMARY KEY (SearchDT, IP\_address),

CONSTRAINT Fk\_user\_srch FOREIGN KEY (Email)

REFERENCES dbo.USERS (Email))

CREATE TABLE dbo.RESULTS (

Name Varchar(80) NOT NULL,

SearchDT Datetime NOT NULL,

IP\_address Varchar(15) NOT NULL,

CONSTRAINT Pk\_result PRIMARY KEY (Name, SearchDT, IP\_address),

CONSTRAINT Fk\_product FOREIGN KEY (Name)

REFERENCES dbo.PRODUCTS (Name),

CONSTRAINT Fk\_search FOREIGN KEY (SearchDT, IP\_address)

REFERENCES dbo.SEARCHES (SearchDT, IP\_address))

CREATE TABLE dbo.SEEDS (

Name Varchar(80) NOT NULL,

Size Char(5) NOT NULL,

Season Varchar(6) NOT NULL,

Sun\_amount Varchar(30) NOT NULL,

CONSTRAINT Pk\_seed PRIMARY KEY (Name),

CONSTRAINT Fk\_Name\_seed FOREIGN KEY (Name)

REFERENCES dbo.PRODUCTS (Name))

CREATE TABLE dbo.SEED\_TYPES (

Name Varchar(80) NOT NULL,

Type Varchar(20) NOT NULL,

CONSTRAINT Pk\_seed\_type PRIMARY KEY (Name, Type),

CONSTRAINT Fk\_seed FOREIGN KEY (Name)

REFERENCES dbo.SEEDS (Name))

CREATE TABLE dbo.GARDENS (

Name Varchar(80) NOT NULL,

Small\_count Tinyint NOT NULL,

Large\_count Tinyint NOT NULL,

CONSTRAINT Pk\_garden PRIMARY KEY (Name),

CONSTRAINT Fk\_Name\_Grdn FOREIGN KEY (Name)

REFERENCES dbo.PRODUCTS (Name))

CREATE TABLE dbo.CHOSENS (

Garden Varchar(80) NOT NULL,

Seed Varchar(80) NOT NULL,

Quantity Tinyint NOT NULL DEFAULT 1,

CONSTRAINT Pk\_chosen PRIMARY KEY (Garden, Seed),

CONSTRAINT Fk\_Garden FOREIGN KEY (Garden)

REFERENCES dbo.GARDENS (Name),

CONSTRAINT Fk\_Seed\_chs FOREIGN KEY (Seed)

REFERENCES dbo.SEEDS (Name))

CREATE TABLE dbo.DETAILS (

Name Varchar(40) NOT NULL,

Address Varchar(150) NOT NULL,

Company Varchar(40) NULL,

Phone# Varchar(30) NOT NULL,

CONSTRAINT Pk\_details PRIMARY KEY (Name, Address))

CREATE TABLE dbo.DETAILS\_OF (

Email Varchar(40) NOT NULL,

Name Varchar(40) NOT NULL,

Address Varchar(150) NOT NULL,

CONSTRAINT Pk\_dtlsOf PRIMARY KEY (Email, Name, Address),

CONSTRAINT Fk\_dtls\_usr FOREIGN KEY (Email)

REFERENCES dbo.USERS (Email),

CONSTRAINT Fk\_detOf\_usr FOREIGN KEY (Name, Address)

REFERENCES dbo.DETAILS (Name, Address))

CREATE TABLE dbo.ORDERS (

OrderID Int NOT NULL,

Email Varchar(40) NULL,

Name Varchar(40) NOT NULL,

Address Varchar(150) NOT NULL,

OrderDate Date NOT NULL,

Shipping\_method Varchar(100) NULL,

Payment\_type Varchar(7) NOT NULL,

CONSTRAINT Pk\_order PRIMARY KEY (OrderID),

CONSTRAINT Fk\_user\_ordr FOREIGN KEY (Email)

REFERENCES dbo.USERS (Email),

CONSTRAINT Fk\_dtls\_ordr FOREIGN KEY (Name, Address)

REFERENCES dbo.DETAILS (Name, Address))

CREATE TABLE dbo.INCLUSIONS (

OrderID Int NOT NULL,

Name Varchar(80) NOT NULL,

Quantity Tinyint NOT NULL DEFAULT 1,

CONSTRAINT Pk\_inclusion PRIMARY KEY (OrderID, Name),

CONSTRAINT Fk\_ordr\_ncl FOREIGN KEY (OrderID)

REFERENCES dbo.ORDERS (OrderID),

CONSTRAINT Fk\_prdc\_ncl FOREIGN KEY (Name)

REFERENCES dbo.PRODUCTS (Name))

# אילוצי מרחב ערכים

**אילוצי בדיקה (CHECK):**

* **אילוצים על Price ו-Discount:** כדי לאכוף שמחירו של מוצר מסוים לא יוגדר להיות שלילי, ושההנחה על מוצר מסוים לא תהיה גדולה ממחירו.

ALTER TABLE dbo.PRODUCTS

ADD CONSTRAINT Ck\_Price

CHECK (Price > 0),

CONSTRAINT Ck\_Discount

CHECK (Discount <= Price)

* **אילוץ על Email:** כדי לאכוף שכתובת מייל של משתמש רשום מסוים תהיה בפורמט תקין.

ALTER TABLE dbo.USERS

ADD CONSTRAINT Ck\_Email

CHECK (Email LIKE '%@%.%')

* **אילוץ על IP\_address:** כדי לאכוף שכתובת IP של פרטי לקוח מסוימים תהיה בפורמט תקין.

ALTER TABLE dbo.SEARCHES

ADD CONSTRAINT Ck\_IP\_address

CHECK ((ParseName(IP\_address, 4) BETWEEN 0 AND 255)

AND(ParseName(IP\_address, 3) BETWEEN 0 AND 255)

AND(ParseName(IP\_address, 2) BETWEEN 0 AND 255)

AND(ParseName(IP\_address, 1) BETWEEN 0 AND 255))

* **אילוצים על Size ו-Season:** כדי לאכוף שערכי השדות הללו הם אחד הערכים התקינים.

ALTER TABLE dbo.SEEDS

ADD CONSTRAINT Ck\_Size

CHECK (Size IN ('Small', 'Large')),

CONSTRAINT Ck\_Season

CHECK (Season IN ('Summer', 'Spring', 'Winter', 'Fall'))

* **אילוצים על Small\_count ו-Large\_count:** כדי לאכוף שערכי השדות הללו הם אחד הערכים התקינים.

ALTER TABLE dbo.GARDENS

ADD CONSTRAINT Ck\_counts

CHECK ((Small\_count = 2 AND Large\_count = 2)

OR (Small\_count = 5 AND Large\_count = 1)

OR (Small\_count = 8 AND Large\_count = 0))

* **אילוצים על Quantity בCHOSENS ובINCLUSIONS:** כדי לאכוף שכמות מסוימת של מוצר מסוים הנכלל במוצר מסוים אחר או הזמנה מסוימת בהכרח חיובי.

ALTER TABLE dbo.CHOSENS

ADD CONSTRAINT Ck\_Quantity\_chs

CHECK (Quantity > 0)

ALTER TABLE dbo.INCLUSIONS

ADD CONSTRAINT Ck\_Quantity\_ncl

CHECK (Quantity > 0)

* **אילוץ על Phone#:** כדי לאכוף שמספר טלפון הוא בפורמט תקין.

ALTER TABLE dbo.DETAILS

ADD CONSTRAINT Ck\_Phone#

CHECK (Phone# LIKE '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9] [0-9][0-9]')

**טבלאות חיפוש (Lookup Tables):**

* **טבלת חיפוש** **ל-Seed\_type:** הוספנו טבלת חיפוש זו מכיוון שכל זרע משויך לקטגוריה אחת או יותר וזה מיוצג על ידי השדה מרובה הערכים Seed\_type, אבל יכול להיווצר המצב בו יוחלט בחברה להוסיף קטגוריה חדשה.

CREATE TABLE dbo.SEEDTYPELOOKUP (

Type Varchar(20) PRIMARY KEY NOT NULL)

INSERT INTO dbo.SEEDTYPELOOKUP

VALUES

('Greens'), ('Herbs'), ('Fruiting'), ('Flowers'), ('Root Vegetables'), ('Heirloom')

ALTER TABLE dbo.SEED\_TYPES

ADD CONSTRAINT Fk\_seedtypeLU FOREIGN KEY (Type)

REFERENCES dbo.SEEDTYPELOOKUP (Type)

* **טבלת חיפוש** **ל-Payment\_type:** הוספנו טבלת חיפוש עבור סוג התשלום מתוך הנחה כי אפשרי ללקוח לשלם רק דרך שירותי התשלום הנמצאים באתר, ובשלב מסוים יכולה החברה להוסיף שירות תשלום נוסף.

CREATE TABLE dbo.PYMNTTYPELOOKUP (

Type Varchar(7) PRIMARY KEY NOT NULL)

INSERT INTO dbo.PYMNTTYPELOOKUP

VALUES

('ShopPay'), ('Klarna')

ALTER TABLE dbo.ORDERS

ADD CONSTRAINT Fk\_Pymnt\_type FOREIGN KEY (Payment\_type)

REFERENCES dbo.PYMNTTYPELOOKUP (Type)

# SCRIPT להזנת רשומות

**הערה:** בטבלת PRODUCTS יש 12 רשומות ובטבלת CHOSENS יש 15 רשומות, זאת כדי לאפשר את החסם התחתון (5) של הרשומות לכל הטבלאות כפי שנדרש בחלק זה של הפרויקט.

INSERT INTO dbo.PRODUCTS

VALUES

('Fennel', 12, 0),

('Scallion', 10, 0),

('Tomato: Heirloom Purple', 10, 0),

('Sunflower: Golden', 12, 0),

('Pepper: Large Bell', 9, 0),

('Light Salsa Garden', 119.99, 20),

('Root & Vegi Salad Garden', 129.99, 25),

('Simple Vegi Salad Garden', 129.99, 25),

('Custom: SnflwrGldn2, PprLB2', 159.99, 30),

('Custom: Scln3, Fnl2, SnflrGldn3', 159.99, 10),

('SUPER Fertilizer', 34.99, 3.75),

('Pruning & Harvesting Scissors', 39.99, 10)

INSERT INTO dbo.RELATIONS

VALUES

('SUPER Fertilizer', 'Light Salsa Garden'),

('SUPER Fertilizer', 'Root & Vegi Salad Garden'),

('SUPER Fertilizer', 'Sunflower: Golden'),

('Pruning & Harvesting Scissors', 'Light Salsa Garden'),

('Pruning & Harvesting Scissors', 'Root & Vegi Salad Garden'),

('Pruning & Harvesting Scissors', 'Fennel'),

('Pruning & Harvesting Scissors', 'Scallion')

INSERT INTO dbo.USERS

VALUES

('tomge@post.bgu.ac.il', 'Mis63677'),

('sophiada@post.bgu.ac.il', 'Mis99988'),

('juliev@post.bgu.ac.il', 'Mis45774'),

('georgebush@gmail.com', 'g01w09B6'),

('mickeyM@walla.co.il', 'Hoho1235')

INSERT INTO dbo.SEARCHES

VALUES

('1999-03-16 13:57:22', '10.100.102.13', 'mickeyM@walla.co.il', 'Seed'),

('2006-04-29 10:30:12', '255.30.2.0', NULL, 'Fertilizer'),

('2013-02-01 14:03:54', '99.234.8.8', 'juliev@post.bgu.ac.il', 'Burger'),

('2015-06-02 15:57:51', '82.43.43.74', 'georgebush@gmail.com', 'Pen'),

('2023-12-30 20:26:24', '69.120.55.26', 'sophiada@post.bgu.ac.il', 'Fennel'),

('2024-01-03 23:56:38', '0.74.255.254', 'sophiada@post.bgu.ac.il', 'Google')

INSERT INTO dbo.RESULTS

VALUES

('Fennel', '1999-03-16 13:57:22', '10.100.102.13'),

('Scallion', '1999-03-16 13:57:22', '10.100.102.13'),

('Tomato: Heirloom Purple', '1999-03-16 13:57:22', '10.100.102.13'),

('Sunflower: Golden', '1999-03-16 13:57:22', '10.100.102.13'),

('Pepper: Large Bell', '1999-03-16 13:57:22', '10.100.102.13'),

('SUPER Fertilizer', '2006-04-29 10:30:12', '255.30.2.0'),

('Fennel', '2023-12-30 20:26:24', '69.120.55.26')

INSERT INTO dbo.SEEDS

VALUES

('Fennel', 'Small', 'Spring', 'Full Sun / Partial Shade'),

('Scallion', 'Small', 'Winter', 'Partial Shade'),

('Tomato: Heirloom Purple', 'Large', 'Summer', 'Full Sun'),

('Sunflower: Golden', 'Small', 'Spring', 'Prefers Full Sun'),

('Pepper: Large Bell', 'Large', 'Fall', 'Prefers Full Sun')

INSERT INTO dbo.SEED\_TYPES

VALUES

('Fennel', 'Greens'),

('Fennel', 'Fruiting'),

('Fennel', 'Root Vegetables'),

('Scallion', 'Herbs'),

('Tomato: Heirloom Purple', 'Heirloom'),

('Tomato: Heirloom Purple', 'Fruiting'),

('Sunflower: Golden', 'Flowers'),

('Pepper: Large Bell', 'Fruiting')

INSERT INTO dbo.GARDENS

VALUES

('Light Salsa Garden', 2, 2),

('Root & Vegi Salad Garden', 5, 1),

('Simple Vegi Salad Garden', 2, 2),

('Custom: SnflwrGldn2, PprLB2', 2, 2),

('Custom: Scln3, Fnl2, SnflrGldn3', 8, 0)

INSERT INTO dbo.CHOSENS

VALUES

('Light Salsa Garden', 'Pepper: Large Bell', 1),

('Light Salsa Garden', 'Tomato: Heirloom Purple', 1),

('Light Salsa Garden', 'Scallion', 2),

('Root & Vegi Salad Garden', 'Fennel', 2),

('Root & Vegi Salad Garden', 'Tomato: Heirloom Purple', 1),

('Root & Vegi Salad Garden', 'Scallion', 3),

('Simple Vegi Salad Garden', 'Tomato: Heirloom Purple', 1),

('Simple Vegi Salad Garden', 'Pepper: Large Bell', 1),

('Simple Vegi Salad Garden', 'Fennel', 1),

('Simple Vegi Salad Garden', 'Scallion', 1),

('Custom: SnflwrGldn2, PprLB2', 'Sunflower: Golden', 2),

('Custom: SnflwrGldn2, PprLB2', 'Pepper: Large Bell', 2),

('Custom: Scln3, Fnl2, SnflrGldn3', 'Sunflower: Golden', 3),

('Custom: Scln3, Fnl2, SnflrGldn3', 'Fennel', 2),

('Custom: Scln3, Fnl2, SnflrGldn3', 'Scallion', 3)

INSERT INTO dbo.DETAILS

VALUES

('Mickey H Mouse the 1st', 'United States, California, Disney Land, , , 00000001', 'Disney', '1234598889'),

('Goofy & Pluto', 'United States, California, Disney Land, , , 00000001', 'Disney', '0126549854'),

('Sargent Donald Duck', 'United States, California, Disney Land, , , 00000001', 'Disney', '1235516595'),

('George W. Bush', 'United States, Texas, Alburkurky, Saint st., 3, 23611356', 'The Government', '2222222221'),

('Laura Welch', 'United States, Texas, Alburkurky, Midland st., 8, 23615656', NULL, '2222222222'),

('Sargent Donald Duck', 'United States, Texas, Alburkurky, Saint st., 3, 23611356', 'The Government', '1235516595')

INSERT INTO dbo.DETAILS\_OF (Email, Name, Address)

VALUES

('mickeyM@walla.co.il', 'Mickey H Mouse the 1st', 'United States, California, Disney Land, , , 00000001'),

('mickeyM@walla.co.il', 'Goofy & Pluto', 'United States, California, Disney Land, , , 00000001'),

('mickeyM@walla.co.il', 'Sargent Donald Duck', 'United States, California, Disney Land, , , 00000001'),

('georgebush@gmail.com', 'George W. Bush', 'United States, Texas, Alburkurky, Saint st., 3, 23611356'),

('georgebush@gmail.com', 'Sargent Donald Duck', 'United States, Texas, Alburkurky, Saint st., 3, 23611356')

INSERT INTO dbo.ORDERS

VALUES

(33222, 'mickeyM@walla.co.il', 'Mickey H Mouse the 1st', 'United States, California, Disney Land, , , 00000001', '1999-03-16', 'Pickup from the factory', 'Klarna'),

(33223, NULL, 'Laura Welch', 'United States, Texas, Alburkurky, Midland st., 8, 23615656', '2009-03-16', NULL, 'Klarna'),

(33224, 'georgebush@gmail.com', 'George W. Bush', 'United States, Texas, Alburkurky, Saint st., 3, 23611356', '2015-08-19', NULL, 'ShopPay'),

(33225, 'mickeyM@walla.co.il', 'Sargent Donald Duck', 'United States, California, Disney Land, , , 00000001', '2016-06-06', 'Fedex', 'ShopPay'),

(33226, 'sophiada@post.bgu.ac.il', 'Goofy & Pluto', 'United States, California, Disney Land, , , 00000001', '2023-12-31', 'Fedex', 'ShopPay'),

(33227, 'georgebush@gmail.com', 'George W. Bush', 'United States, Texas, Alburkurky, Saint st., 3, 23611356', '2024-01-05', 'Fedex', 'Klarna'),

(33228, NULL, 'Laura Welch', 'United States, Texas, Alburkurky, Midland st., 8, 23615656', '2024-02-03', 'Private jet delivery', 'ShopPay')

INSERT INTO dbo.INCLUSIONS

VALUES

(33222, 'Custom: SnflwrGldn2, PprLB2', 3),

(33222, 'Custom: Scln3, Fnl2, SnflrGldn3', 3),

(33223, 'SUPER Fertilizer', 2),

(33224, 'Pruning & Harvesting Scissors', 1),

(33225, 'Light Salsa Garden', 20),

(33226, 'Fennel', 60),

(33227, 'Pruning & Harvesting Scissors', 5),

(33228, 'Sunflower: Golden', 30)

# SCRIPT להסרת טבלאות ואילוצים

ALTER TABLE dbo.INCLUSIONS

DROP CONSTRAINT Pk\_inclusion, Fk\_ordr\_ncl, Fk\_prdc\_ncl, Ck\_Quantity\_ncl

DROP TABLE dbo.INCLUSIONS

ALTER TABLE dbo.ORDERS

DROP CONSTRAINT Pk\_order, Fk\_user\_ordr, Fk\_dtls\_ordr, Fk\_Pymnt\_type

DROP TABLE dbo.ORDERS

DROP TABLE dbo.PYMNTTYPELOOKUP

ALTER TABLE dbo.DETAILS\_OF

DROP CONSTRAINT Pk\_dtlsOf, Fk\_dtls\_usr, Fk\_detOf\_usr

DROP TABLE dbo.DETAILS\_OF

ALTER TABLE dbo.DETAILS

DROP CONSTRAINT Pk\_details, Ck\_Phone#

DROP TABLE dbo.DETAILS

ALTER TABLE dbo.CHOSENS

DROP CONSTRAINT Pk\_chosen, Fk\_Garden, Fk\_Seed\_chs, Ck\_Quantity\_chs

DROP TABLE dbo.CHOSENS

ALTER TABLE dbo.GARDENS

DROP CONSTRAINT Pk\_garden, Fk\_Name\_Grdn, Ck\_counts

DROP TABLE dbo.GARDENS

ALTER TABLE dbo.SEED\_TYPES

DROP CONSTRAINT Pk\_seed\_type, Fk\_seed, Fk\_seedtypeLU

DROP TABLE dbo.SEED\_TYPES

DROP TABLE dbo.SEEDTYPELOOKUP

ALTER TABLE dbo.SEEDS

DROP CONSTRAINT Pk\_seed, Fk\_Name\_seed, Ck\_Size, Ck\_season

DROP TABLE dbo.SEEDS

ALTER TABLE dbo.RESULTS

DROP CONSTRAINT Pk\_result, Fk\_product, Fk\_search

DROP TABLE dbo.RESULTS

ALTER TABLE dbo.SEARCHES

DROP CONSTRAINT Pk\_search, Fk\_user\_srch, Ck\_IP\_address

DROP TABLE dbo.SEARCHES

ALTER TABLE dbo.USERS

DROP CONSTRAINT Pk\_user, Ck\_Email

DROP TABLE dbo.USERS

ALTER TABLE dbo.RELATIONS

DROP CONSTRAINT Pk\_relation, Fk\_Product1, Fk\_Product2

DROP TABLE dbo.RELATIONS

ALTER TABLE dbo.PRODUCTS

DROP CONSTRAINT Pk\_product, Ck\_Price, Ck\_Discount

DROP TABLE dbo.PRODUCTS