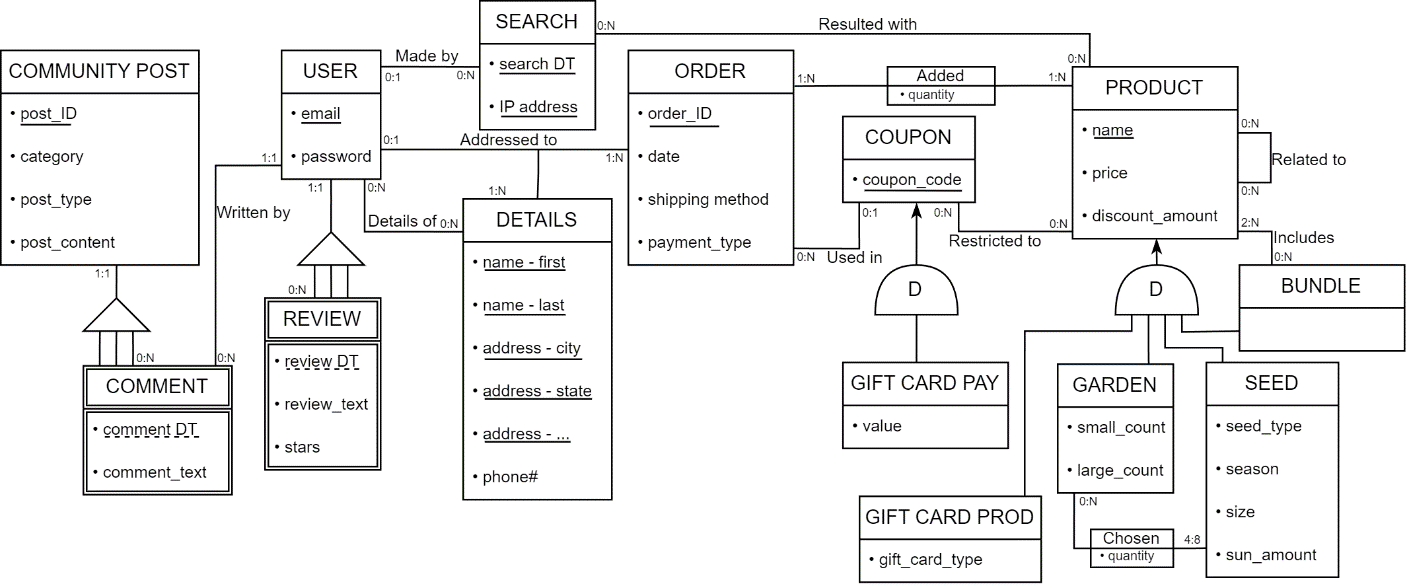
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| מס' קבוצה | אתר | | | תאריך הגשה |
| 16 | [seedsheets.com](http://seedsheets.com) | | | 19/02/2024 |
| חברי הצוות – מספרי ת.ז | | | | |
| 318334901 | | 211680509 | 318255882 | |

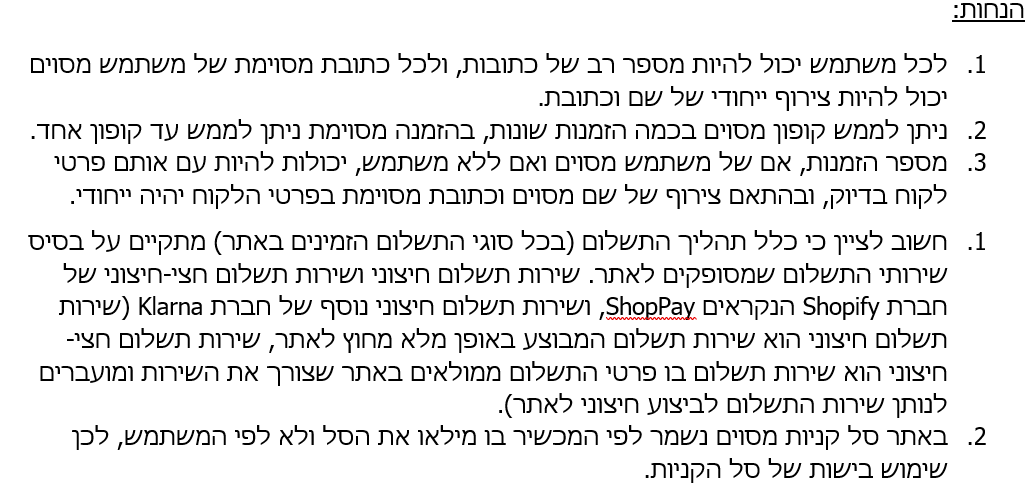
## פרויקט בסיסי נתונים

## מטלה מקדימה – ה-ERD שהוגש בחלק א'

**התרשים בהגשה הקודמת**



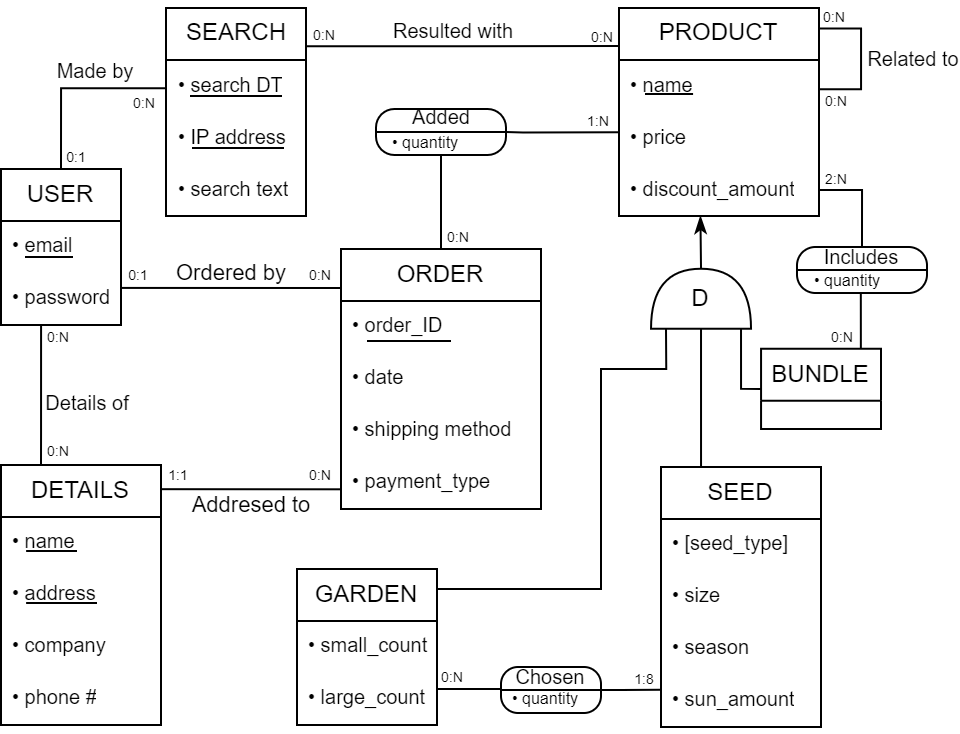
**ההנחות שנרשמו בהגשה הקודמת**



**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   * search DT * IP address * Search text | * מפתח ראשי: צירוף השדות search DT, 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, משתמש רשום מסוים או משתמש מזדמן מסוים 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 כאחת. * 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 כאחת. * מפתח ראשי: צירוף השדות name, address – כל צירוף של שם מלא מסוים וכתובת מסוימת באופן ייחודי באתר כך שלכל צירוף של שם מלא מסוים וכתובת מסוימת יהיה שם חברה אחד ומספר טלפון אחד שיכול להירשם בשדות המתאימים | כפי שמצוין בהנחות הקשר Details of למשתמש רשום מסוים יכולות להיות שמורות אינספור פרטי לקוח הנשמרים ברשימת הכתובות של המשתמש, בכל כתובת ברשימה זו ניתן לשים צירוף שונה לחלוטין של הכתובת והשם של הלקוח |
| הקשר "Details of" בין USER לDETAILS | * פרטי לקוח מסוימים יכולים להיות שמורים אצל אינספור משתמשים רשומים שונים או לאף משתמש רשום. ולכן הקרדינליות של קשר זה בצד הישות USER היא 0:N. * למשתמש רשום מסוים יכולים להיות שמורים אינספור פרטי לקוח או אף לא אחת. ולכן הקרדינליות של קשר זה בצד הישות DETAILS היא 0:N. |  |
| Entity: ORDER   * order\_ID * date * shipping method * payment\_type | * מפתח ראשי: order\_ID – מספר סידורי מזהה של הזמנה מסוימת אשר לא יחזור על עצמו. * payment\_type – שדה המכיל את שם שירות התשלום החיצוני שבו השתמש הלקוח בעת התשלום על ההזמנה. * הזמנה שנעשית ע"י משתמש לא רשום לא תשמור כתובת אימייל ויהיה שימוש במספר הטלפון השמור בפרטי הלקוח המצוינים להזמנה כאמצעי התקשורת עם הלקוח. * הקשר "Addressed to" – הזמנה יכולה להיעשות ע"פ פרטי לקוח יחידים בלבד. לכן הקרדינליות של קשר זה בצד הישות DETAILS היא 1:1. פרטי לקוח מסוימים יכולים להיות מצוינים באינספור הזמנות או אף לא אחת. לכן הקרדינליות של קשר זה בצד הישות ORDER היא 0:N. * הקשר "Ordered by" – כל הזמנה מסוימת יכולה להיעשות ע"י לכל היותר משתמש רשום אחד או אף לא אחד. לכן הקרדינליות של קשר זה בצד הישות USER היא 0:1. כל משתמש רשום מסוים יכול לעשות אינספור הזמנות או אף לא אחת. לכן הקרדינליות של קשר זה בצד הישות ORDER היא 0:1. | 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\_amount | * 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\_amount – 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 מ0. * 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 "Added" 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: BUNDLE | * ישות בת במודל ההורשה של ישות מוצר, אשר אין לה תכונות נוספות. * לעסקת חבילה יש מחיר והנחה שלא תלויה במחירם של המוצרים הנכללים בה. |  |
| הקשר "Includes" בין BUNDLE וPRODUCT   * quantity | * כל מוצר מסוים יכול להיכלל באינספור עסקאות חבילה או באף אחת מהן. לכן הקרדינליות של קשר זה בצד הישות BUNDLE היא 0:N. * בכל עסקת חבילה מסוימת יכולים להיכלל אינספור מוצרים, אך לכל הפחות שניים. לכן הקרדינליות של קשר זה בצד הישות PRODUCT היא 2:N. * quantity – כמות של מוצר מסוים שנכלל בעסקת חבילה 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] – שדה מרובה ערכים שמייצג את סוג הצמח, תחת איזה קטגוריה/קטגוריות של צמח/צמחים הצמח הזה יושב, צמח מסוים יכול להיות מסווג כחלק מכמה משפחות של סוגי צמחים. * size – גודל החלקה שהזרע, וכתוצאה מהזרע הצמח, לוקח. יכול להיות או גדול או קטן. * season, sun\_amount – תכונות המכילות מידע אינפורמטיבי על הזרע והצמח המוצגות ללקוח באתר. |  |
| Child Entity: GARDEN   * small\_count * large\_count | * ישות בת במודל ההורשה של ישות מוצר. * קיימות באתר גינות בעיצוב אישי וגינות מוכנות מראש אשר כמות החלקות מהגדלים השונים זהה ואף הזרעים שנכללים בגינה זהים אך מחירם שונה, זאת כתוצאה מעיצוב אישי של גינה באופן שזהה לגינה מוכנה מראש. * small\_count, large\_count – תכונות המכילות מספרים ערך שלם לא שלילי, באופן ספציפי התכונה small\_count גדולה מ0. תכונות אלו מייצגות את סוגי החלקות בגינה מבחינת סדר הגודל שלהן. |  |
| 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 של מוצר מסוים שנכלל בעסקת חבילה מסוימת. שדה זה יכיל מספרים שלמים גדולים מ0. * בגינה מסוימת יכללו בדיוק כמות הזרעים שהיא יכולה להכיל וזה לפי הכמויות וסוגי החלקות המצוינים בישות הגינה המסוימת. |  |

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

**< כאן מוצג פירוט חלקי של הטבלאות. יש לכלול בהגשה רשימה מלאה של הטבלאות והשדות, כולל סימון מתאים של מפתחות ראשיים וזרים >**

* **REGISTERED COSTUMERS** (Email, First\_Name, Last\_Name, Age, Gender)
* **SEARCHES** (Search\_DT, IP\_Address, Origin, Destination, Start\_Date, End\_Date, People\_No, Customer\_Email (REGISTERED COSTUMER)
* **ORDERS** (Order\_No, Total\_Price, Details, Email, Name, Phone No., Country, Paid\_By (PAYMENTS), {Search\_DT, IP\_Address} (SEARCHES))

**…**

**הנחות והסברים נוספים:**

**< כאן מוצג פירוט חלקי. יש לכלול בהגשה רשימה מלאה של הסברים רלוונטיים להבנת המודל הטבלאי – למשל, הנחות נרמול נוספות, הסבר מדוע נבחר מודל מסוים כאשר ניתן היה לממש בכמה צורות שונות, וכד'>**

* **ביטוי קשר בין SEARCH ל-ORDER** – קשר יחיד ליחיד. על פי חוקי המעבר למודל הטבלאי יכולנו לבחור האם לבטא את הקשר ב-SEARCH או ב-ORDER. בחירה ב-ORDER מוסיפה לנו 2 שדות נוספים (מכיוון שהמזהה של SEARCH מורכב מ-2 שדות) ובחירה ב-SEARCH מותירה המון ערכי NULL. בצומת החלטה זו החלטנו כי ערכי ה-NULL המרובים יהוו מכשול ובעייתיות בתחקור מסד הנתונים, ולכן הוחלט לבטא את הקשר בטבלת ה-ORDER.
* ...

# פירוט שדות בסיס הנתונים והערכת גודלם

**< כאן מוצג פירוט חלקי. יש לכלול בהגשה את כלל הטבלאות והשדות>**

| **Table** | **Field** | **Data Type** | **Size (Bytes)** |
| --- | --- | --- | --- |
| **REGISTERED COSTUMER** | Email | Varchar(40) | 40 |
| First\_Name | Varchar(20) | 20 |
| Last\_Name | Varchar(20) | 20 |
| Age | Int | 4 |
| Gender | Char | 1 |
| **Total Record Size (Bytes)** | | **85** |
| **SEARCHES** | Search\_DT | Date Time | 8 |
| IP\_Address | Varchar(15) | 15 |
| Origin | Varchar(40) | 40 |
| Destination | Varchar(40) | 40 |
| Start\_Date | Date | 3 |
| End\_Date | Date | 3 |
| People\_No | Int | 4 |
| Customer\_Email **(REGISTERED COSTUMER**) | Varchar(40) | 40 |
| **Total Record Size (Bytes)** | | **153** |
| **…** |  | |  |

**הנחות והסברים נוספים:**

**< כאן מוצג פירוט חלקי. במידת הצורך יש לכלול בהגשה הסברים רלוונטיים להבנת תכנון השדות >**

* שדות התאריכים בטבלה SEARCHES נבחרו כתאריך ולא כתאריך/זמן, כיוון שהאתר אינו מאפשר חיפוש על פי שעות ספציפיות.

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

# אילוצי מרחב ערכים

**אילוצי בדיקה (CHECK):**

**< כאן מוצגת דוגמא אחת. יש לכלול בהגשה את כלל האילוצים בהתאם לנדרש במטלה, בליווי הקוד>**

* **אילוץ על EMAIL**: הוספנו אילוץ על השדה Email לטבלת בהן מופיע שדה מסוג זה. זהו שדה חשוב מכיוון שהלקוח צריך לקבל את כל פרטי ההזמנה לכתובת המייל שהזין. כמו כן, דרך המייל החברה מזהה את הלקוח באופן חד ערכי ומתקשרת עימו. שדה זה ייחודי עבור כל לקוח ולכן עלינו לבדוק את תקינותו.

CONSTRAINT CK\_Email CHECK (Email like '%@%.%')

**טבלאות חיפוש (Lookup Tables):**

**< כאן מוצגת דוגמא אחת. יש לכלול בהגשה את כלל טבלאות החיפוש בהתאם לנדרש במטלה, בליווי הקוד>**

* **טבלת חיפוש** **ל-Payments:** הוספנו טבלת חיפוש עבור סוג כרטיס האשראי (Type) מתוך הנחה כי לקוח יכול לשלם רק בכרטיסי אשראי מורשים אשר איתם עובדת החברה.

CREATE TABLE CreditTypes (

Type Varchar (20) PRIMARY KEY

)

INSERT INTO CreditLOOKUP VALUES

('VISA'), ('MasterCard'), ('JCB'), ('AmericanExpress'), ('UniunPay-CreditCard')

ALTER TABLE Payments

ADD CONSTRAINT FK\_credittype FOREIGN KEY (type) REFERENCRS CreditTypes (type)

# SCRIPT לבניית הטבלאות

**< כאן מוצגת דוגמא אחת. יש לכלול בהגשה את הקוד של כלל פקודות יצירת הטבלאות>**

CREATE TABLE Registered\_customers (

Email Varchar (40) not null,

First\_Name Varchar (20) not null,

Last\_Name Varchar (20) not null,

Age Int null,

Gender Char(1) null,

CONSTRAINT PK\_customers PRIMARY KEY (Email),

CONSTRAINT ck\_email CHECK (Email like '%@%.%'),

CONSTRAINT ck\_gender CHECK (Gender in ('M','F'))

)

# SCRIPT להזנת רשומות

**< כאן מוצגת דוגמא אחת. יש לכלול בהגשה את הקוד של כלל פקודות ההזנה הרלוונטיות>**

INSERT INTO Registered\_customers (Email, First\_Name, Last\_Name, Age, Gender)

VALUES

('abc@gmail.com', 'Guy', 'Fried', 23, 'M'),

('david\_levi@yahoo.co.il', 'David', 'Levi', 45, 'M'),

('eyal.golan@walla.co.il', 'Eyal', 'Golan', 50, 'M'),

('israelisrali@gmail.com', 'Israel', 'Israeli', 30, 'M'),

('noa\_kile@noakile.domain.com', 'Noa', 'Kirl', 18, 'F'),

# SCRIPT להסרת טבלאות ואילוצים

**< כאן מוצגת דוגמא אחת. יש לכלול בהגשה את הקוד של כלל פעולות ההסרה הרלוונטיות>**

ALTER TABLE Registered\_customers

DROP CONSTRAINT PK\_customers, CK\_email, CK\_gender

DROP TABLE Registered\_customers