



# מערכות קבצים ומסדי נתונים

## פרויקט הגשה – פרויקט תכנות

### מגישים

ת.ז. : 204397244

ת.ז. : 312493000

שם מלא סטודנט 1 : יהונתן כובאני

שם מלא סטודנט 2 : דניאל מודילבסקי

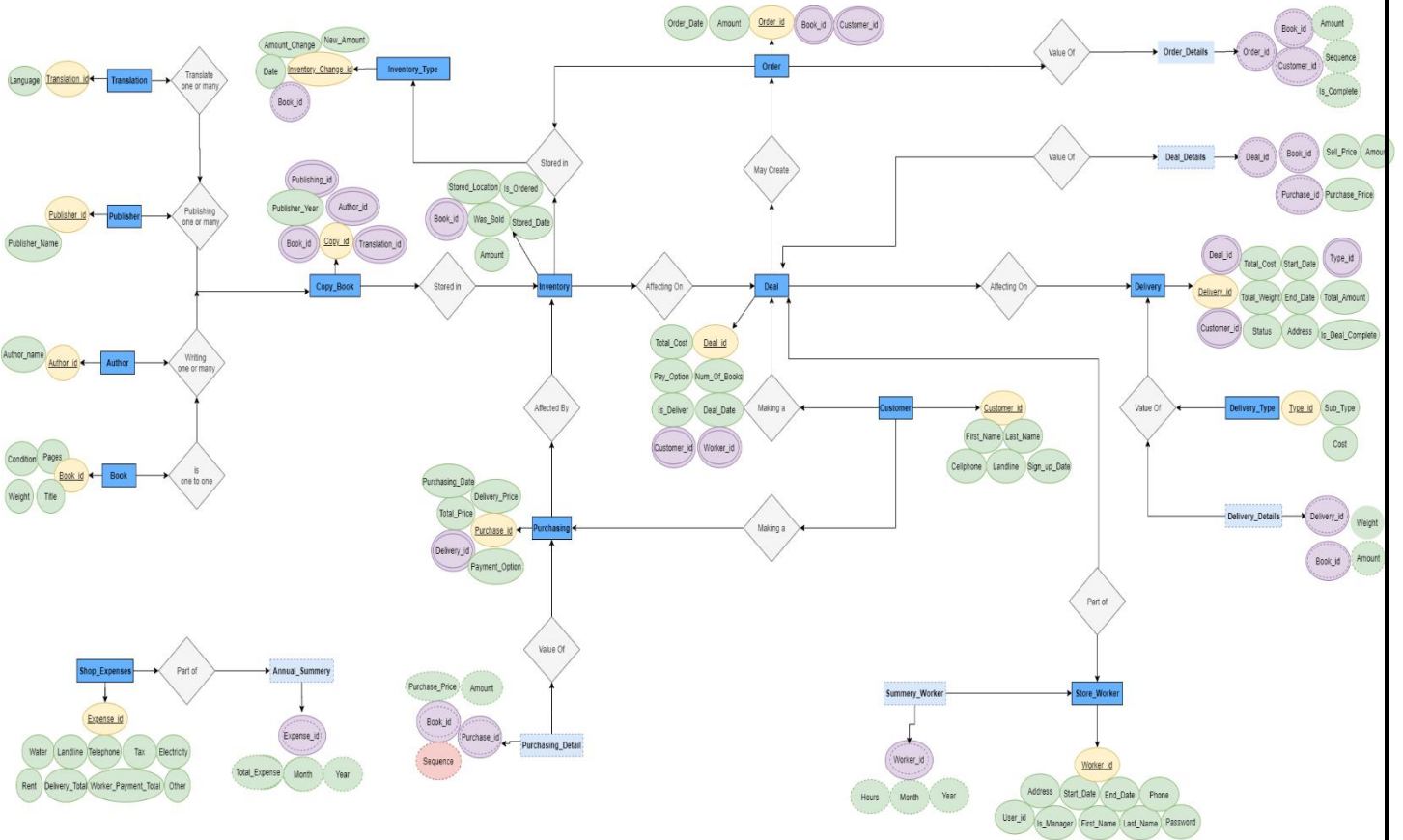
## תוכן עניינים

03.....	<b>1. מידול מסד הנתונים במודל ER</b>
03.....	- 1.1 - תרשים ER נוכחי
03.....	- 1.2 - מקרא לדיאגרמה
04.....	- 1.3 - תרשים דיאגרמת מחלקות נוכחי
05.....	- 1.4 - מפתחות
06.....	- 1.5 - הסברים על הדיאגרמה
06.....	• 1.5.1 - ישויות שהם PK
06.....	• 1.5.2 - קבוצות ישויות חלשות
07.....	• 1.5.3 - סוגי קשרים
08.....	<b>2. כלל ההנחות לצורך יצירת תרשים ה- ERD</b>
06.....	<b>3. רשימת תלויות פונקציונליות</b>
10.....	<b>4. סכמה ראשונית</b>
10.....	- 4.1 - שמות טבלאות
11.....	- 4.2 - מבנה כל טבלה - שמות עמודות, טיפוסים
12.....	- 4.3 - מפתחות
13.....	<b>5. סכמה של בסיס הנתונים לאחר נירמול ל- 3NF</b>
13.....	- 5.1 - הסבר כללי
14.....	- 5.2 - נירמול טבלאות תוכן
14.....	- 5.3 - נירמול טבלאות סוג
15.....	- 5.4 - נירמול מתאמים
16.....	- 5.5 - נירמול אובייקטים עצמאיים
17.....	- 5.6 - סכמה
19.....	<b>6. תוכן כל הטבלאות</b>
26.....	<b>7. מסמך שאילתות</b>
33.....	<b>8. מסמך אלגברת יחסים</b>

## 1. מידול מסד הנתונים במודל ER

### 1.1 - תרשים דיאגרמת ER נוכחי :

• הערה – מצורף בזאת קובץ נלווה – "Diagram-ERD.PNG"



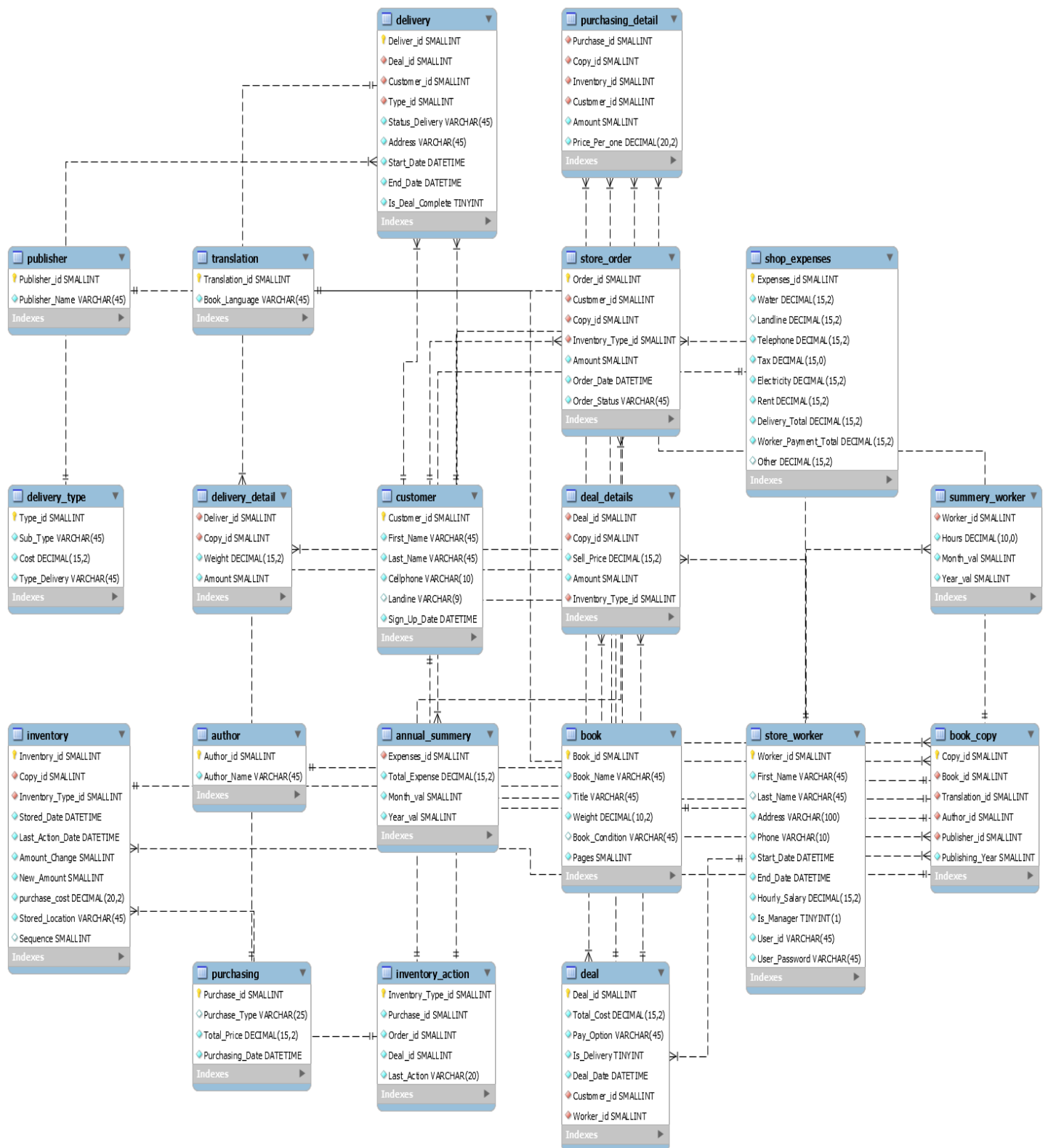
### 1.2 - מקרא סימונים

#### מקרא סימונים:

מלבן=ישות.	אליפסה=תכונות הישות.	אליפסה בתוך אליפסה= ערכית-תכונה מרובת ערכים.	חלשה- ישות שאין לה מפתח-לה קיום עצמאי.	מעוין=קשר בינארי בין ישויות.	קו תחת-מפתח ראשי של טבלה (Primary Key).
שם הישות	תכונה	שם הישות	ילד של עובד	מחלקה	תלידה

## 1.2 - תרשים דיאגרמת מחלקות נוכחי :

• הערה – מצורף בזאת קובץ נלווה – "Class-Diagram.PDF"



#### 1.4 – מפתחות

- הערה – כל הקשרים שהם foreign key הינם קשרי CASCADE גם על Update וגם על Delete.

#### 1.5.1 – קבוצת ישויות שהם PK

מקור	סוג מפתח	יעד	שדה	סוג קשר	סיבה
סופר	Primary Key	/	/	/	אובייקט עצמאי - ניצור ממנו רשימת אובייקטים ללא תלות באחרים
תרגום	Primary Key	/	/	/	אובייקט עצמאי - ניצור ממנו רשימת אובייקטים ללא תלות באחרים
הוצאה לאור	Primary Key	/	/	/	אובייקט עצמאי - ניצור ממנו רשימת אובייקטים ללא תלות באחרים
ספר	Primary Key	/	/	/	אובייקט עצמאי - ניצור ממנו רשימת אובייקטים ללא תלות באחרים
עותק ספר	Primary Key	/	/	/	אובייקט ייחודי - כל שורה היא חח"ע
עובד	Primary Key	/	/	/	אובייקט עצמאי - ניצור ממנו רשימת אובייקטים ללא תלות באחרים
הוצאות	Primary Key	/	/	/	אובייקט עצמאי - ניצור ממנו רשימת אובייקטים ללא תלות באחרים
רכש	Primary Key	/	/	/	אובייקט ייחודי - כל שורה היא חח"ע
הזמנה	Primary Key	/	/	/	אובייקט ייחודי - כל שורה היא חח"ע
עסקה	Primary Key	/	/	/	אובייקט ייחודי - כל שורה היא חח"ע
מלאי	Primary Key	/	/	/	אובייקט ייחודי - כל שורה היא חח"ע
תנועות במלאי	Primary Key	/	/	/	טבלה קבועה - ללא שינויים בערכים
משלוח	Primary Key	/	/	/	אובייקט ייחודי - כל שורה היא חח"ע
סוג משלוח	Primary Key	/	/	/	טבלה קבועה - ללא שינויים בערכים

#### 1.5.2 – קבוצת ישויות חלשות

מקור	סוג מפתח	יעד	שדה	סוג קשר	סיבה
רכש פרטים	/	/	/	/	קבוצת ישויות חלשה - ללא PK יכול להכיל שדות כפולים ועובד ביחס רבים ליחיד
הזמנה פרטים	/	/	/	/	קבוצת ישויות חלשה - ללא PK יכול להכיל שדות כפולים ועובד ביחס רבים ליחיד
משלוח פריטים	/	/	/	/	קבוצת ישויות חלשה - ללא PK יכול להכיל שדות כפולים ועובד ביחס רבים ליחיד
עסקה פריטים	/	/	/	/	קבוצת ישויות חלשה - ללא PK יכול להכיל שדות כפולים ועובד ביחס רבים ליחיד
סיכום חודשי עובד	/	/	/	/	קבוצת ישויות חלשה - ללא PK יכול להכיל שדות כפולים ועובד ביחס רבים ליחיד
סיכום חודשי חנות	/	/	/	/	קבוצת ישויות חלשה - ללא PK יכול להכיל שדות כפולים ועובד ביחס רבים ליחיד

- 1.5.3 – קבוצת ישויות חלשות

- 1.4 – מפתחות

- 3 – רשימת תלויות פונקציונליות

- הערה – טבלת פתרון זאת רלוונטית לשלושת הסעיפים הבאים :

מקור	סוג מפתח	יעד	שדה	סוג קשר	סיבה
סופר	Foreign Key	עותק ספר	Author_id	One to One	לעותק ספר יש סופר שכתב אותו
תרגום	Foreign Key	עותק ספר	Translation_id	One to One	לעותק ספר יש תרגום שבו הוא כתוב
הוצאה לאור	Foreign Key	עותק ספר	Publisher_id	One to One	לעותק ספר יש הוצאה לאור שמוציאה אותו לאור
ספר	Foreign Key	עותק ספר	Book_id	One to One	יצירת ערך לספר שיכיל פרטים של אחד מ-4 הנתונים הנ"ל, וכל שינוי של נתון (ויש המון מבחינה עסקית) יקבל עותק ספר משל עצמו עם id ייחודי
עובד	Foreign Key	סיכום חודשי עובד	Customer_id	One to One	מניעת כפילויות של נתוני עובד על ידי הפרדה בין הנתונים שלו שלא משתנים בתדירות לשעות העבודה שלו שכן משתנים בתדירות
הוצאות חנות	Foreign Key	סיכום חודשי חנות	Expenses_id	One to One	מניעת כפילויות של נתוני עובד על ידי הפרדה בין הנתונים שלה שלא משתנים בתדירות להוצאות החודשיות שלה שכן משתנות בתדירות
לקוח	Foreign Key	רכש פרטים	Customer_id	One to One	רכישה של החנות מתבצעת כשהספק הוא הלקוח
מלאי	Foreign Key	רכש פרטים	Inventory_id	One to One	הרכישה של החנות נכנסת למלאי החנות
רכש עותק ספר	Foreign Key	רכש פרטים	Purchase_id	One to many	על רכישה אחת יכולים להיות כמה פריטים
	Foreign Key	רכש פרטים	Copy_id	One to One	רכישה מכילה עותקי ספרים

מקור	סוג מפתח	יעד	שדה	סוג קשר	סיבה
עותק ספר	Foreign Key	הזמנה	Copy_id	One to One	הזמנה היא לספרים
תנועות במלאי	Foreign Key	הזמנה	Inventory_Type_id	One to One	הזמנה מושפעת מהאם הספר חוזר למלאי
לקוח	Foreign Key	הזמנה	Customer_id	One to One	הזמנה מבוצעת על ידי לקוח
הזמנה	Foreign Key	הזמנה פרטים	Order_id	One to many	על הזמנה אחת יכולים להיות כמה פריטים
לקוח	Foreign Key	עסקה	Customer_id	One to One	לקוח מבצע עסקה בחנות
עובד	Foreign Key	עסקה	Worker_id	One to One	עובד מטפל בעסקת הלקוח
עסקה	Foreign Key	עסקה פרטים	Deal_id	One to many	על עסקה אחת יכולים להיות כמה פריטים
תנועות במלאי	Foreign Key	עסקה פרטים	Inventory_Type_id	One to One	עסקה מושפעת מהאם הספר קיים במלאי
עותק ספר	Foreign Key	עסקה פרטים	Copy_id	One to One	עסקה מורכבת מעותקים של ספרים
תנועות במלאי	Foreign Key	מלאי	Inventory_Type_id	One to One	סיווג איזו פעולה אחרונה בוצעה על מלאי זה
עותק ספר	Foreign Key	מלאי	Copy_id	One to One	מלאי מורכב מעותקים של ספרים
סוג משלוח	Foreign Key	משלוח	Type_id	One to One	משלוח יכול להיות אחד מ-5 סוגים שונים בתעריפים שונים
עסקה	Foreign Key	משלוח	Deal_id	One to many	על עסקה אחת יכולים לצאת כמה משלוחים
לקוח	Foreign Key	משלוח	Customer_id	One to One	משלוח מגיע לכתובת שהזין הלקוח
משלוח	Foreign Key	משלוח פריטים	Deliver_id	One to many	על משלוח אחד יכולים להיות מספר פריטים
עותק ספר	Foreign Key	משלוח פריטים	Copy_id	One to One	משלוח מורכב מעותקים של ספרים

## 2. כלל ההנחות לצורך יצירת תרשים ה-ERD

- **הערה כללית** – כל אובייקט נבנה על פי דרישות הפרויקט – בא לידי ביטוי בשדות כל פרויקט ובדרישות העסקיות הנובעות ממנו.
- **מימד זמן** – מחייב הוספת מימד זמן כלשהו לרוב המתאמים והפעולות על מנת להבדיל ביניהם ברמת הביצוע.
- **רשימות עצמאיות** – נפריד גופים אשר הם ללא תלות אחד בשני, וניצור להם אובייקט שיהיה בסיס גם להרחבה של עוד שדות בהמשך במידה והחנות תרצה בכך. זאת על מנת לא ליצור עומס ממקוד בטבלאות מסוימות.

### - **עותק ספר**

- יכולים להיות המון עותקים של ספרים שההבדל ביניהם הוא לכל הפחות אחד.
- לספר יכולים להיות מספר מחברים או מספר תרגומים או מספר הוצאות לאור לכן נחזיק עותק ספר לכל אפשרות שכזאת.

### - **מלאי**

- מחזיק כמויות של ספרים.
- מקשר בין פעילויות החנות הנוגעות בכמויות הספרים והימצאותן.
- תיעוד פעולה אחרונה – מלאי מתעדכן בכל פעולת "תנועת מלאי" שמתבצעת ותמיד ישמור את ההשפעה של הפעולה האחרונה (קיים ערך sequence) שמתעד מספר פעולות שנעשו על עותק מסויים במלאי.
- יכול להחזיק מלאי 0 מעותק של ספר, אך לא יכול להיות מלאי שלילי – מראה האם ספר מסוים היה אי פעם במלאי.
- עותק שלא קיים במלאי יקבל שורה חדשה, עותק שקיים יתעדכן בכמות ובסוג הפעולה.
- מושפע מרכש, עסקה, הזמנה.
- חלק מהספרים הם בחנות וחלק במחסן.

### - **רכש**

- נעשה או מחנות שנסגרת או מלקוח, יבוא לידי ביטוי בשדה של סוג רכש.

### - **תנועות במלאי**

- הצגה של איזו פעולה מתבצעת עכשיו על המלאי – טבלה קבועה של ערכים.

### - **רכישה**

- פעולה עצמאית – תחזיק id אחד



#### - הזמנה

- פעולה עצמאית – תחזיק id אחד
- מותר להזמין פריט פר הזמנה.
- סטאטוס – בעת עדכון שחזר ערך מסוים למלאי, החנות מבצעת עדכון (update) לסעיף סטאטוס של ההזמנה – דבר המציג את התהליך מאחורי הקלעים שעושה החנות כדי לסגור מעגל ולטפל בהזמנת לקוח.

#### - עסקה

- פעולה עצמאית – תחזיק id אחד

#### - משלוח

- סוג - יקבע לפי טבלה קבועה של סוג משלוח ועלות. W
- עלות – קיימת עלות כוללת ובנוסף יחושב בנפרד עלות משלוח המושפעת מסוג משלוח ומשקל המשלוח.
- כמה משלוחים יכולים להחזיק עסקה אחת – מראה שאפשר על עסקה אחת להוציא יותר ממשלוח אחד (פיצול משלוחים).
- יוצא רק על עסקאות שסומן עליהם כן על משלוח, יש אפשרות לקנות במקום.

#### - לקוח

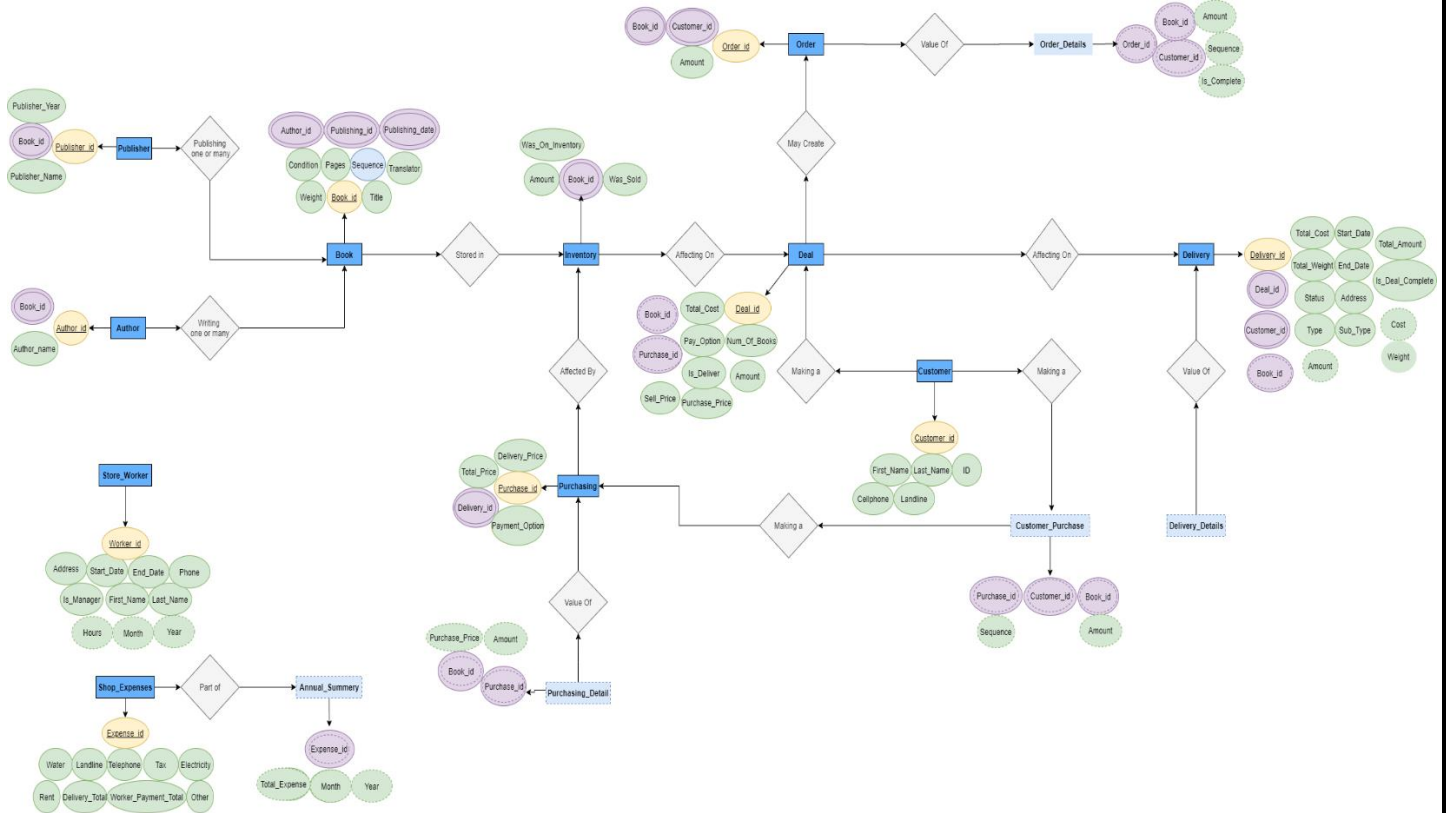
- גורם עצמאי – יחזיק id אחד
- טלפון פרטי - ייחודי

#### - עובד

- גורם עצמאי – יחזיק id אחד
- עוהד יכול להיות מנהל.
- שם משתמש – ייחודי
- טלפון פרטי – ייחודי
- תיעוד של עובדים שכבר לא עובדים בחנות על ידי מימד זמן – תאריך התחלה וסוף.

#### 4. סכמה ראשונית

##### 4.1 – סכמה - שמות טבלאות



##### 4.2 – מבנה כל טבלה – שמות עמודות, טיפוסים

###### 1. Author – סופר

Author_id	Author_Name	Book_id
-----------	-------------	---------

###### 2. Publisher – הוצאה לאור

Publisher_id	Publisher_Name	Book_id	Publisher_Year
--------------	----------------	---------	----------------

###### 3. Translation – תרגום

Translation_id	Translation_Language	Book_id
----------------	----------------------	---------

###### 4. Book – ספר

Book_id	Book_Name	Title	Weight	Book_Condition	Pages
Translation_id	Author_id	Publisher_id	Publishing_date	sequence	

###### 5. Customer – לקוח

Customer_id	First_Name	Last_Name	Cellphone	Landline	Sign_Up_Date
-------------	------------	-----------	-----------	----------	--------------

## 6 . Purchasing – רכש

Purchase_id	Payment_Option	Total_Price	Purchasing_Date
-------------	----------------	-------------	-----------------

## 7 . Purchasing\_Detail – רכש פריטים

Purchase_id	Book_id	Amount	Price_Per_one
-------------	---------	--------	---------------

## 8 . Inventory – מלאי

Book_id	Amount	Was_On_Inventory	Was_Sold
---------	--------	------------------	----------

## 9 . Store\_Worker – עובד

Worker_id	First_Name	Last_Name	Address	Phone	--
Start_Date	End_Date	Hours	Is_Manager	Month	Year

## 10 . Shop\_Expenses – הוצאות

Expenses_id	Water	Landline	Telephone	Tax
Electricity	Rent	Delivery_Total	Worker_Payment_Total	Other

## 11 . Annual\_Summery – סיכום חנות חודשי

Expenses_id	Total_Expense	Month_val	Year_val
-------------	---------------	-----------	----------

## 12 . Deal – עסקה

Deal_id	Sell_Price	Pay_Option	Is_Delivery	Deal_Date
Book_id	Purchas_id	Num_Books	Worker_id	Customer_id

## 12 . Store\_Order – הזמנה

Order_id	Customer_id	Book_id	Amount	Order_Date	Order_Status
----------	-------------	---------	--------	------------	--------------

## 13 . Delivery – משלוח

Deliver_id	Deal_id	Customer_id	Type	Sub_Type
Status_Delivery	Address	Start_Date	End_Date	Is_Deal_Complete
Book_id	Total_Cost	Cost	Weight	Amount

## 14 . Customer\_Purchase – רכישת לקוח

Book_id	Customer_id	Purchase_id	Amount
---------	-------------	-------------	--------

4.3 – מפתחות

מקור	סוג מפתח	יעד	שדה	סוג קשר
סופר	Primary Key	/	/	/
תרגום	Primary Key	/	/	/
הוצאה לאור	Primary Key	/	/	/
ספר	Primary Key	/	/	/
עובד	Primary Key	/	/	/
הוצאות	Primary Key	/	/	/
רכש	Primary Key	/	/	/
הזמנה	Primary Key	/	/	/
עסקה	Primary Key	/	/	/
משלוח	Primary Key	/	/	/
רכש פרטים	/	/	/	/
סופר	Foreign Key	ספר	Author_id	One to One
תרגום	Foreign Key	ספר	Translation_id	One to One
הוצאה לאור	Foreign Key	ספר	Publisher_id	One to One
ספר	Foreign Key	סופר	Book_id	One to One
ספר	Foreign Key	תרגום	Book_id	One to One
ספר	Foreign Key	הוצאה לאור	Book_id	One to One
לקוח	Foreign Key	רכש פרטים	Customer_id	One to One
ספר	Foreign Key	רכש פרטים	Book_id	One to One
ספר	Foreign Key	הזמנה	Book_id	One to One
לקוח	Foreign Key	הזמנה	Customer_id	One to One
לקוח	Foreign Key	עסקה	Customer_id	One to One
עובד	Foreign Key	עסקה	Worker_id	One to One
ספר	Foreign Key	מלאי	Book_id	One to One
עסקה	Foreign Key	משלוח	Deal_id	One to many
לקוח	Foreign Key	משלוח	Customer_id	One to One
ספר	Foreign Key	משלוח	Book_id	One to One

## 5. סכמה של בסיס הנתונים לאחר נירמול ל-3NF

- 5.1 – **הסבר כללי** – תבנית R עם קבוצת תלויות F החלה עליה היא בצורה נורמאלית 3NF אם כל תלות  $X \rightarrow A$  בסגור של F מקיים לפחות את אחד משלושת התנאים הבאים:

- 5.1.1 –  $X \rightarrow A$  – היא טריוויאלית – A מוכל ב-X.
- 5.1.2 – X – מפתח על.
- 5.1.3 – A – הוא חלק ממפתח של R.

- 5.2 – **נירמול טבלאות תוכן**

• **צורך** – מניעת כפילות של נתונים, יצירת מציאות של הכלת נתונים כך שלאובייקט המכיל יש שדה מפתח יחיד ולו יכולים להיות מספר פריטים שונים שמשווייכים אליו.

• **עסקה – מצב לפני נירמול** – על אובייקט אחד של עסקה היו המון שורות של פריטים (במידה ולקוח קנה מעל פריט אחד), כל פריט קיבל ID משלו, נוצר קושי בניהול עסקאות, שיוך פריט לעסקה עצמה ובניהול מסד הנתונים.

• **עסקה – מצב אחרי נירמול** – נוצרה הפרדה בין עסקה שהיא אובייקט X לבין "פריט עסקה" שהם אובייקט A (סעיף 5.1.1). הפרדה זאת נותנת יכולת של הכלה בה לעסקה אחת עם ID אחד יש מספר פריטים שונים **המשוייכים** לאותה עסקה, **ללא כפל בנתונים**, ללא סתירה ונותן יכולת קלה יותר בניהול מסד הנתונים.

• **רכש – מצב לפני נירמול** – על אובייקט אחד של רכש היו המון שורות של פריטים (במידה וחנות רוכשת מעל פריט אחד), כל פריט קיבל ID משלו, נוצר קושי בניהול רכישות, שיוך פריט לרכישה עצמה ובניהול מסד הנתונים.

• **רכש – מצב אחרי נירמול** – נוצרה הפרדה בין רכישה שהיא אובייקט X לבין "פריט רכישה" שהם אובייקט A (סעיף 5.1.1). הפרדה זאת נותנת יכולת של הכלה בה לרכישה אחת עם ID אחד יש מספר פריטים שונים **המשוייכים** לאותה רכישה, **ללא כפל בנתונים**, ללא סתירה ונותן יכולת קלה יותר בניהול מסד הנתונים.

• **משלוח – מצב לפני נירמול** – על אובייקט אחד של משלוח היו המון שורות של פריטים (במידה ולקוח מזמין מעל פריט אחד או מוציא מעל משלוח אחד לאותה עסקה), כל פריט קיבל ID משלו, נוצר קושי בניהול משלוחים, שיוך פריט למשלוח ובניהול מסד הנתונים.

• **משלוח – מצב אחרי נירמול** – נוצרה הפרדה בין משלוח שהוא אובייקט X לבין "פריט משלוח" שהם אובייקט A (סעיף 5.1.1). הפרדה זאת נותנת יכולת של הכלה בה למשלוח אחד עם ID אחד יש מספר פריטים שונים **המשוייכים** לאותו משלוח, **ללא כפל בנתונים**, ללא סתירה ונותן יכולת קלה יותר בניהול מסד הנתונים.

### - 5.3 – נירמול טבלאות סוג

- **צורך** – מניעת בזבוז נפח מסד הנתונים, ומניעת סרבול עדכון של נתונים שלא מתעדכנים והינם קבועים.
- **משלוח** – **מצב לפני נירמול** – לכל משלוח היום 3 עמודות נוספות של סוג משלוח תת סוג משלוח ועלות שהינם נתוהים קבועים ויצאו סירבול ועומס ל-DB ככל שנוספו שורות תוכן.
- **משלוח** – **מצב אחרי נירמול** – הוצאת 3 הנתונים הקבועים לטבלה אחת והקישור בינה לבין משלוח על ידי מפתח זר מה שמצמם חזרתיות של נתונים ובזבוז זכרון מקון.
- **תנועות מלאי** – **מצב לפני נירמול** – לכל פעולה שקשורה למלאי הייתה נוספת שורה חדשה עם אותו מספר עותק ספר סוג הפעולה והשינוי. מה שיצר עומס רב על טבלת המלאי, אי סדר בכמויות המלאי ובהשפעות עליו.
- **תנועות מלאי** – **מצב אחרי נירמול** – קיטלוג אפשרויות ההשפעה על המלאי ל-3 אופציות והוצאתן החוצה מהמלאי כולל ההשפעה על הכמות. מצב שבו אין כפילות יותר בשורות בטבלת המלאי אלא עדכון הכמות הנוכחית + סוג הפעולה האחרונה שבוצעה. בשיטה זאת מונעים סירבול מסד הנתונים ומקבלים תמונה עדכנית תמיד של כמות המלאי לכל עותק ספר ומה הפעולה האחרונה שהושפע ממנה.

### - 5.4 – נירמול מתאמים

- **צורך** – מניעת כפילות של נתונים וסתירה של אמינות הטבלה מאי התאמות בין טבלאות שונות, הקלה בהליכי עדכון ומחיקה.
- **עותק** – **מצב לפני נירמול** – לא היה קיים, הייתה כפילות של ספרים, וחוסר סדר בחיבור בין ספר, עורך, הוצאה לאור, ותרגום ביחסים של אחד לרבים לדוגמה ספר אחד עם מספר עורכים.
- **עותק** – **מצב אחרי נירמול** – ייחדוי ובעל מפתח, מתאם בין ארבעת האובייקטים ויוצר הפרדה בין כל סוג וסוג תוך אפשרות לייצר המון סוגי עותקים שונים.
- **מלאי** – **מצב לפני נירמול** – לא היה קיים, היה עומס בשדות שונים עבור ספר, שגם ככה היה עמוס מחזרתיות וכפילות של שורות משילוב עם אובייקטים אחרים כמו עורך.
- **מלאי** – **מצב אחרי נירמול** – ייחדוי ובעל מפתח, מתאם בין סוגי עותקים לכל פעולה המשפיעה על המלאי ויוצר הפרדה בין כל סוג וסוג תוך אפשרות לייצר המון סוגי עותקים שונים ולנהל אותם מבחינה כמותית.

## 5.5 – רשימה עצמאית

- **צורך** – מניעת בעיות כפילות נתונים, מניעת נפח זיכרון מיותר ויצירת ערכים עצמאיים ללא תלות שתדירות השינוי שלהם מאוד נמוכה.
- **עובד** – **מצב לפני נירמול** – עובד היה מכיל פרטי עובד ובנוסף פרטי שעות חודשיות (נתון שהיה מתעדכן בתדירות גדולה) ויצר שורות רבות ומיותרות של נתונים קבועים של עובד מסוים.
- **עובד** – **מצב אחרי נירמול** – יצירת רשימת עובדים ללא תלות באף גורם אחר, ועליהן נבנה מתאמים על מנת לקשר בינן לבין אובייקטים אחרים כמו שעות.
- **לקוח** – **מצב לפני נירמול** – לקוח היה מכיל פרטי לקוח ובנוסף פרטי רכישות שביצע (נתון שהיה מתעדכן בתדירות לא קטנה) ויצר שורות רבות ומיותרות של נתונים קבועים של לקוח מסוים.
- **לקוח** – **מצב אחרי נירמול** – יצירת רשימת לקוחות ללא תלות באף גורם אחר, ועליהן נבנה מתאמים על מנת לקשר בינן לבין אובייקטים אחרים כמו ספר.
- **עורך** – **מצב לפני נירמול** – עורך היה מכיל פרטי עורך ובנוסף פרטי ספרים שאותם כתב (נתון שהיה מתעדכן בתדירות גדולה) ויצר שורות רבות ומיותרות של נתונים קבועים של עורך מסוים.
- **עורך** – **מצב אחרי נירמול** – יצירת רשימת עורכים ללא תלות באף גורם אחר, ועליהן נבנה מתאמים על מנת לקשר בינן לבין אובייקטים אחרים כמו ספר.
- **תרגום** – **מצב לפני נירמול** – תרגום היה מכיל פרטי תרגום ובנוסף פרטי ספרים שמתורגמים לשפה זו (נתון שהיה מתעדכן בתדירות גדולה) ויצר שורות רבות ומיותרות של נתונים קבועים של עורך מסוים.
- **תרגום** – **מצב אחרי נירמול** – יצירת רשימת תרגומים ללא תלות באף גורם אחר, ועליהן נבנה מתאמים על מנת לקשר בינן לבין אובייקטים אחרים כמו ספר.
- **הוצאה לאור** – **מצב לפני נירמול** – הוצאה לאור הייתה מכילה פרטי הוצאה לאור ובנוסף פרטי ספרים שהוציאה (נתון שהיה מתעדכן בתדירות גדולה) מה שיצר שורות רבות ומיותרות של נתונים קבועים של הוצאה לאור מסוימת.
- **הוצאה לאור** – **מצב אחרי נירמול** – יצירת רשימת הוצאות לאור ללא תלות באף גורם אחר, ועליהן נבנה מתאמים על מנת לקשר בינן לבין אובייקטים אחרים כמו ספר.
- **ספר** – **מצב לפני נירמול** – ספר היה מכיל פרטי ספר ובנוסף פרטי עורך, תרגום, הוצאה לאור שקשורים לספר (נתון שהיה קיים ביחס אחד לרבים) ויצר שורות רבות ומיותרות של נתונים קבועים של ספר מסוים.
- **ספר** – **מצב אחרי נירמול** – יצירת רשימת ספרים ללא תלות באף גורם אחר, ועליהן נבנה מתאמים על מנת לקשר בינן לבין אובייקטים אחרים כמו עותק של ספר.

- **הוצאות חנות – מצב לפני נירמול** – הוצאות חנות היו מכילות פרטי חשבונות ובנוסף הוצאות לפי חודש (נתון שהיה קיים ביחס אחד לרבים) ויצר שורות רבות ומיותרות של נתונים קבועים של הוצאות החנות.
- **הוצאות חנות – מצב אחרי נירמול** – יצירת רשימת הוצאות חודשית ללא תלות באף גורם אחר, ועליהן נבנה מתאמים על מנת לקשר בין לבין אובייקטים אחרים כמו עותק של סוג ההוצאות.



## 5.6 - סכמה לאחר נירמול 3NF

### 1. Author – סופר

Author_id	Author_Name
-----------	-------------

### 2. Publisher – הוצאה לאור

Publisher_id	Publisher_Name
--------------	----------------

### 3. Translation – תרגום

Translation_id	Translation_Language
----------------	----------------------

### 4. Book – ספר

Book_id	Book_Name	Title	Weight	Book_Condition	Pages
---------	-----------	-------	--------	----------------	-------

### 5. Book\_Copy – עותק מספר

Copy_id	Book_id	Translation_id	Author_id	Publisher_id	Publishing_Year
---------	---------	----------------	-----------	--------------	-----------------

### 6. Customer – לקוח

Customer_id	First_Name	Last_Name	Cellphone	Landline	Sign_Up_Date
-------------	------------	-----------	-----------	----------	--------------

### 7. Purchasing – רכש

Purchase_id	Purchase_Type	Total_Price	Purchasing_Date
-------------	---------------	-------------	-----------------

### 8. Purchasing\_Detail – רכש פריטים

Purchase_id	Copy_id	Inventory_id	Customer_id	Amount	Price_Per_one
-------------	---------	--------------	-------------	--------	---------------

### 9. Inventory – מלאי

Inventory_id	purchase_cost	Inventory_Type_id	Stored_Date	Copy_id
Amount_Change	New_Amount	Last_Action_Date	Stored_Location	--

### 10. Inventory\_Action – תנועות במלאי

Inventory_Type_id	Purchase_id	Order_id	Deal_id	Last_Action
-------------------	-------------	----------	---------	-------------

### 11. Store\_Worker – עובד

Worker_id	First_Name	Last_Name	Address	Phone	--
Start_Date	End_Date	Hourly_Salary	Is_Manager	User_id	User_Password

12 . Summery\_Worker – סיכום חודשי עובד

Worker_id	Hours	Month_val	Year_val
-----------	-------	-----------	----------

13 . Shop\_Expenses – הוצאות

Expenses_id	Water	Landline	Telephone	Tax
Electricity	Rent	Delivery_Total	Worker_Payment_Total	Other

14 . Annual\_Summery – סיכום חנות חודשי

Expenses_id	Total_Expense	Month_val	Year_val
-------------	---------------	-----------	----------

15 . Deal – עסקה

Deal_id	Total_Cost	Pay_Option	Is_Delivery	Deal_Date	Customer_id	Worker_id
---------	------------	------------	-------------	-----------	-------------	-----------

16 . Deal\_Details – עסקה פרטים

Deal_id	Copy_id	Sell_Price	Amount	Inventory_Type_id
---------	---------	------------	--------	-------------------

17 . Store\_Order – הזמנה

Order_id	Customer_id	Copy_id	Inventory_Type_id	Amount	Order_Date	Order_Status
----------	-------------	---------	-------------------	--------	------------	--------------

18 . Delivery\_Type – סוג משלוח

Type_id	Sub_Type	Cost	Type_Delivery
---------	----------	------	---------------

19 . Delivery – משלוח

Deliver_id	Deal_id	Customer_id	Type_id	--
Status_Delivery	Address	Start_Date	End_Date	Is_Deal_Complete

20 . Delivery\_Detail – משלוח פרטים

Deliver_id	Copy_id	Weight	Amount
------------	---------	--------	--------

## 6. תוכן כל הטבלאות

## annual\_summery

	Expenses_id	Total_Expense	Month_val	Year_val
▶	1	120067.00	3	2020
	2	167254.00	8	2017
	3	182990.00	11	2014
	4	102222.00	12	2006
	5	123321.00	1	1997
	6	98000.00	4	2001
	7	164223.00	9	2019
	8	129678.00	2	2015
	9	345543.00	2	2012
	10	456654.00	1	2008

## author

	Author_id	Author_Name
▶	1	Daniel Silvan
	2	Judi Pico
	3	Harlen Coban
	4	Ram Oren
	5	Halen Huserny
	6	Gabi Nirzan
	7	J K Rolling
	8	Benny Goren
	9	Yohel Geva
	10	Henery Bill
•	NULL	NULL

## Book

	Book_id	Book_Name	Title	Weight	Book_Condition	Pages
▶	1	Dangerous trap	thriller	2.20	Brand new	321
	2	Badolina	thriller	2.20	Brand new	321
	3	Rose Efect	Roman	2.20	Brand new	321
	4	Rich Father Poor Father	Business	2.20	Brand new	200
	5	My sister Guard	Comedy	1.60	dilapidated	125
	6	De Vin Chi Code	thriller	3.50	dilapidated	456
	7	Do not leave	thriller	2.20	medium	321
	8	Harry Potter 1	fantasy	2.20	medium	321
	9	Harry Potter 2	fantasy	2.20	as new	322
	10	Harry Potter 3	fantasy	2.20	as new	323
	11	Abaldarit	thriller	2.20	good	321
	12	Vertigo	thriller	2.20	good	321
	13	Math 5 point	Learning	3.40	Brand new	321
✱	NULL	NULL	NULL	NULL	NULL	NULL

## book\_copy

F	E	D	C	B	A	▶
Publishing_Year	Publisher_id	Author_id	Translation_id	Book_id	Copy_id	
2000	1	1	1	1	1	2
2001	2	1	1	1	2	3
2002	1	3	1	1	3	4
2003	1	1	4	1	4	5
2004	1	2	1	1	5	6
2005	2	1	5	1	6	7
2000	2	2	2	2	7	8
2001	2	1	1	2	8	9
2002	3	3	3	2	9	10
2003	1	1	4	1	10	11
2004	1	7	1	1	11	12
2005	2	6	5	1	12	13
2000	3	3	3	3	13	14
2001	3	3	5	3	14	15
2002	3	3	3	4	15	16
2003	5	2	4	3	16	17
2004	1	2	4	1	17	18
2005	2	5	5	1	18	19
2000	1	6	1	2	19	20
2001	2	1	1	3	20	21
2002	1	3	1	4	21	22
2003	1	1	4	5	22	23
2004	1	2	1	6	23	24
2005	2	1	5	7	24	25
2000	5	5	5	8	25	26
2001	7	7	4	9	26	27
2002	6	6	6	10	27	28
2003	1	1	4	11	28	29
2004	1	2	1	12	29	30
2005	2	1	5	13	30	31

## Customer

	Customer_id	First_Name	Last_Name	Cellphone	Landline	Sign_Up_Date
	1	None	None	0	0	2020-01-01 00:00:00
▶	2	Daniel	Model	507939887	35749494	2019-09-13 00:00:00
	3	Anar	Djafarov	544886618	37584736	2020-06-21 00:00:00
	4	Omri	Haham	526999779	72093847	2019-02-04 00:00:00
	5	Aviram	Shemesh	527247877	49482674	2018-07-29 00:00:00
	6	Lucas	Argentini	586753353	48213123	2019-08-17 00:00:00
	7	Inbal	Model	501112221	85785794	2019-09-13 00:00:00
	8	Halel	Sarusi	507483622	69102854	2018-12-11 00:00:00
	9	Ariel	Lunenfeld	522222222	31111111	2019-10-19 00:00:00
	10	Beni	Benyamin	536040300	36040300	2020-04-04 00:00:00
	11	Ilana	Shapiro	5356789009	41487533	2020-05-29 00:00:00
*	NULL	NULL	NULL	NULL	NULL	NULL

## Deal

	Deal_id	Total_Cost	Pay_Option	Is_Delivery	Deal_Date	Customer_id	Worker_id
▶	1	200.00	Credit	1	2020-07-25 00:00:00	3	2
	2	200.00	Credit	1	2020-07-25 00:00:00	4	2
	3	200.00	Credit	1	2020-07-25 00:00:00	2	2
	4	200.00	bank transfer	1	2020-07-24 00:00:00	2	3
	5	200.00	bank transfer	1	2020-07-24 00:00:00	5	3
	6	200.00	bank transfer	1	2020-07-24 00:00:00	6	3
	7	200.00	Bit	1	2020-07-21 00:00:00	7	4
	8	200.00	Bit	0	2020-07-24 00:00:00	8	4
	9	200.00	Bit	0	2020-07-22 00:00:00	3	5
	10	200.00	Bit	1	2020-07-23 00:00:00	2	5
	11	200.00	Bit	1	2020-06-20 00:00:00	2	5
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

## deal\_details

	Deal_id	Copy_id	Sell_Price	Amount	Inventory_Type_id
▶	1	2	45.00	10	1
	1	1	45.00	10	1
	2	3	45.00	10	1
	2	8	45.00	10	1
	3	4	45.00	10	1
	3	5	45.00	10	1
	3	2	45.00	10	1
	4	3	45.00	10	1
	5	6	45.00	10	1
	5	7	45.00	10	1

## Delivery

	Deal_id	Customer_id	Type_id	Status_Delivery	Address	Start_Date	End_Date	Is_Deal_Complete
▶	1	3	1	In_Preparation	Shenkar	2020-02-21 00:00:00	2020-02-22 00:00:00	0
	1	3	2	Delivered	Shenkar	2020-02-21 00:00:00	2020-02-22 00:00:00	1
	2	4	3	In_Preparation	Shenkar	2020-02-21 00:00:00	2020-02-22 00:00:00	0
	3	2	4	Awaiting_Collection	Shenkar	2020-02-21 00:00:00	2020-02-22 00:00:00	0
	4	2	5	Shipped	Shenkar	2020-02-21 00:00:00	2020-02-22 00:00:00	0
	5	5	4	Arrived	Shenkar	2020-02-21 00:00:00	2020-02-22 00:00:00	0
	5	5	4	Awaiting_Collection	Shenkar	2020-02-21 00:00:00	2020-02-22 00:00:00	0
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

## delivery\_detail

	Deliver_id	Copy_id	Weight	Amount
▶	1	1	3.00	3
	1	2	4.00	4
	2	3	5.00	4
	2	6	6.00	1
	3	4	6.00	9
	3	5	7.00	38
	3	2	8.00	7
	4	3	3.00	6
	5	6	2.00	3
	5	7	1.00	5

## delivery\_type

	Type_id	Sub_Type	Cost	Type_Delivery
▶	1	Collection_Point	1.50	Courier_Company Xpress
	2	Courier_Home	2.00	Courier_Company Xpress
	3	Fast_Courier	3.00	Israel_Post
	4	Courier	2.50	Israel_Post
	5	Registered	3.50	Israel_Post
*	NULL	NULL	NULL	NULL

## inventory

Sequence	Stored_Location	purchase_cost	New_Amount	Amount_Change	Last_Action_Date	Stored_Date	Inventory_Type_id	Copy_id	Inventory_id	1
1	Store	25	10	-20	25/07/2020 00:00	25/07/2020 00:00	1	1	1	2
1	Store	25	20	-10	25/07/2020 00:00	25/07/2020 00:00	1	2	2	3
1	Store	25	0	-30	24/07/2020 00:00	25/07/2020 00:00	1	3	3	4
1	Store	25	20	-10	25/07/2020 00:00	25/07/2020 00:00	1	4	4	5
1	Store	25	20	-10	25/07/2020 00:00	25/07/2020 00:00	1	5	5	6
1	Store	25	20	-10	25/07/2020 00:00	25/07/2020 00:00	1	6	6	7
1	Store	25	20	-10	25/07/2020 00:00	25/07/2020 00:00	1	7	7	8
1	Store	25	30	30	25/07/2020 00:00	25/07/2020 00:00	1	8	8	9
1	Store	25	30	30	25/07/2020 00:00	25/07/2020 00:00	1	9	9	10
1	Store	25	30	30	25/07/2020 00:00	25/07/2020 00:00	1	10	10	11
1	Store	25	30	30	25/07/2020 00:00	25/07/2020 00:00	1	11	11	12
1	Store	25	30	30	25/07/2020 00:00	25/07/2020 00:00	1	12	12	13
1	Store	25	30	30	25/07/2020 00:00	25/07/2020 00:00	1	13	13	14
1	Store	25	30	30	25/07/2020 00:00	25/07/2020 00:00	1	14	14	15
1	Store	25	20	-10	26/07/2020 00:00	25/07/2020 00:00	1	15	15	16
1	Store	25	20	-10	26/07/2020 00:00	25/07/2020 00:00	1	16	16	17
1	Stock	25	30	30	25/07/2020 00:00	25/07/2020 00:00	1	17	17	18
1	Stock	25	30	30	25/07/2020 00:00	25/07/2020 00:00	1	18	18	19
1	Stock	25	30	30	25/07/2020 00:00	25/07/2020 00:00	1	20	19	20
1	Stock	25	30	30	02/08/2020 00:00	25/07/2020 00:00	1	21	20	21

## inventory\_action

	Inventory_Type_id	Purchase_id	Order_id	Deal_id	Last_Action
▶	1	0	0	1	Sell
	2	0	1	0	Order_Complete
	3	1	0	0	Purchase
✱	NULL	NULL	NULL	NULL	NULL

## Publisher

	Publisher_id	Publisher_Name
▶	1	Keter
	2	Modan
	3	Yediot Aharonot
	4	Meter
	5	Dvash
	6	Am Oved
	7	Kineret
✱	NULL	NULL

## Purchasing

	Purchase_id	Purchase_Type	Total_Price	Purchasing_Date
▶	1	Closing Store	200.00	2020-07-25 00:00:00
	2	Closing Store	200.00	2020-07-25 00:00:00
	3	Closing Store	200.00	2020-07-25 00:00:00
	4	Customer	200.00	2020-07-25 00:00:00
	5	Customer	200.00	2020-07-25 00:00:00
	6	Customer	200.00	2020-07-25 00:00:00
✱	NULL	NULL	NULL	NULL

## Translation

	Translation_id	Book_Language
▶	1	Hebrew
	2	English
	3	Spanish
	4	German
	5	Russian
	6	Arab
*	NULL	NULL

## purchasing\_detail

	Purchase_id	Copy_id	Inventory_id	Customer_id	Amount	Price_Per_one
▶	1	1	1	1	30	10.00
	1	2	2	1	30	12.00
	1	3	3	1	30	13.00
	1	4	4	1	30	14.00
	1	5	5	1	30	15.00
	2	6	6	1	30	16.00
	2	7	7	1	30	15.00
	2	8	8	1	30	14.00
	3	9	9	1	30	12.00
	3	10	10	1	30	10.00
	4	11	11	2	30	12.00
	4	12	12	2	30	10.00
	4	13	13	2	30	18.00
	4	20	14	2	30	20.00
	5	14	15	3	30	10.00
	5	15	16	3	30	11.00
	5	16	17	3	30	12.00
	6	17	18	4	30	14.00
	6	18	19	4	30	6.00



## shop\_expenses

	Expenses_id	Water	Landline	Telephone	Tax	Electricity	Rent	Delivery_Total	Worker_Payment_Total	Other
▶	1	200.00	200.00	200.00	200	200.00	200.00	1000.00	20000.00	0.00
	2	100.00	100.00	100.00	100	100.00	100.00	1000.00	19000.00	0.00
	3	300.00	300.00	300.00	300	300.00	300.00	1000.00	18000.00	0.00
	4	400.00	400.00	400.00	400	400.00	400.00	1000.00	17000.00	100.00
	5	500.00	500.00	500.00	500	500.00	500.00	1000.00	16000.00	0.00
	6	200.00	200.00	200.00	200	200.00	200.00	1000.00	15000.00	0.00
	7	200.00	200.00	400.00	200	200.00	400.00	1000.00	14000.00	0.00
	8	200.00	400.00	200.00	200	400.00	200.00	1000.00	13000.00	0.00
	9	300.00	300.00	300.00	300	300.00	300.00	1000.00	12000.00	0.00
	10	500.00	500.00	500.00	500	500.00	500.00	1000.00	11000.00	0.00
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

## store\_order

	Order_id	Customer_id	Copy_id	Inventory_Type_id	Amount	Order_Date	Order_Status
▶	1	6	19	3	10	2020-07-27 00:00:00	Waiting
	2	2	15	3	10	2020-07-27 00:00:00	in_Storage
	3	3	16	3	10	2020-07-27 00:00:00	in_Storage
	4	4	10	3	10	2020-07-27 00:00:00	Complete
	5	5	11	3	10	2020-07-27 00:00:00	Complete
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

## store\_worker

User_Password	User_id	Is_Manager	Hourly_Salary	End_Date	Start_Date	Phone	Address	Last_Name	First_Name	Worker_id
Britni123	12345	1	50	12/12/2025 00:00	20/03/2018 00:00	541344678	Tel Aviv	Spirs	Britni	1
Jhonatan	23456	0	30	12/12/2025 00:00	20/04/2019 00:00	512345691	Haifa	Cubani	Jhonatan	2
Daniel123	34578	0	35	12/12/2025 00:00	21/01/2020 00:00	514345666	Jerusalem	Modilevsky	Daniel	3
Anar1234	45678	0	40	12/12/2025 00:00	22/11/2018 00:00	522345655	Jerusalem	Djafarov	Anar	4
Yaniv123	56789	0	35	12/12/2025 00:00	23/12/2020 00:00	532545644	Metula	Zlotnik	Yaniv	5
Omri1234	98765	0	37	12/12/2025 00:00	24/07/2019 00:00	542345633	Metula	Haham	Omri	6
Machloof	87654	0	41	12/12/2025 00:00	25/07/2018 00:00	552345622	Ramat Gan	Machloof	Daniel	7
Irada1234	76543	0	35	12/12/2025 00:00	26/08/2020 00:00	525345611	Ramat Gan	Keren	Irada	8
Muhamad1	65432	0	38	12/12/2025 00:00	13/09/2019 00:00	514345600	Ashdod	Kadafi	Muhamad	9
Mick1234	54321	0	33	12/12/2025 00:00	09/10/2020 00:00	562845634	Ashdod	Tyson	Mick	10

## summery\_worker

	Worker_id	Hours	Month_val	Year_val
▶	1	120	2	2020
	2	140	2	2020
	3	160	2	2020
	4	180	2	2020
	5	200	2	2020
	1	200	3	2020
	2	180	3	2020
	3	150	3	2020
	4	110	3	2020
	5	130	3	2020

## 7. מסמך שאלות

-- 1 --

```

Select b.Book_Name , i.New_Amount as Amount, i.Stored_Location from
Inventory i
inner join Book_Copy BC on BC.Book_id = i.Copy_id
inner join book b on b.Book_id = BC.Book_id
where b.Book_Name = "Badolina" group by i.Sequence;

```

-- 2 --

```

Select * from Customer c order by c.Sign_Up_Date limit 1;

```

-- 3 --

```

select i.Copy_id, i.Stored_Date, b.Book_Name from Inventory i
inner join Book_Copy BC on BC.Book_id = i.Copy_id
inner join book b on b.Book_id = BC.Book_id order by i.Stored_Date
limit 1;

```

-- 4 --

```

select * from Store_Order SO where Order_Status != "Complete"
order by SO.Order_Date;

```

-- 5 --

```

select sum(dd.Amount) as NUm_Of_Books from Deal d
inner join Deal_Details dd on d.Deal_id = dd.Deal_id
inner join Book_Copy BC on BC.Book_id = dd.Copy_id
inner join book b on b.Book_id = BC.Book_id
where b.Book_Name = "Badolina"
group by dd.Amount;

```

-- 6 --

```
select sum(dd.Amount) as Selling_Books , a.Author_Name from Deal d
inner join Deal_Details dd on d.Deal_id = dd.Deal_id
inner join Book_Copy BC on dd.Copy_id = BC.Copy_id
inner join book b on b.Book_id = BC.Book_id
inner join author a on a.Author_id = BC.Author_id
where d.Deal_Date >= "2000-01-01" AND d.Deal_Date <= "2021-01-01"
group by A.Author_Name order by sum(dd.Amount) DESC LIMIT 1;
```

-- 7 --

```
select sum(dd.Amount) as Buing_Books, CONCAT(c.First_Name, " ",
c.Last_Name ) as Customer_Name from Deal d
inner join Deal_Details dd on d.Deal_id = dd.Deal_id
inner join Book_Copy BC on dd.Copy_id = BC.Copy_id
inner join book b on b.Book_id = BC.Book_id
inner join customer c on d.customer_id = c.customer_id
group by d.Customer_id order by count(d.customer_id) DESC limit 3;
```

-- 8 --

```
select count(distinct(BC.Translation_id)) Translation_Num, b.Book_Name from
Inventory i
inner join Book_Copy BC on i.Copy_id = BC.Copy_id
inner join book b on b.Book_id = BC.Book_id
inner join translation t on t.Translation_id = BC.Translation_id
group by b.Book_id order by count(distinct(BC.Translation_id)) DESC limit 1;
```

-- 9 --

```
select CONCAT(c.First_Name, " ", c.Last_Name) as Customer_Name ,
b.book_name, dd.sell_price, d.Deal_Date from Deal d
inner join Deal_Details dd on d.Deal_id = dd.Deal_id
inner join Book_Copy BC on dd.Copy_id = BC.Copy_id
inner join book b on b.Book_id = BC.Book_id
inner join customer c on d.customer_id = c.customer_id
where c.First_Name = "Daniel" And c.Last_Name = "Model"
order by d.Deal_Date;
```

-- 10 --

```
select CONCAT(c.First_Name, " ", c.Last_Name) as Customer_Name,
SO.Order_Date, b.Book_Name, b.Title, b.Book_Condition, SO.Amount,
So.Order_Status from Store_Order SO
inner join Book_Copy BC on SO.Copy_id = BC.Copy_id
inner join book b on b.Book_id = BC.Book_id
inner join customer c on SO.customer_id = c.customer_id
where c.First_Name = "Daniel" And c.Last_Name = "Model"
order by SO.Order_Date;
```

-- 11 --

```
select d.Deliver_id, sum(Amount * dd.Weight * dt.Cost) as Delivery_Price,
CONCAT(c.First_Name, " ", c.Last_Name) as Customer_Name from delivery
d
inner join Delivery_Detail dd on d.Deliver_id = dd.Deliver_id
inner join Delivery_Type dt on d.Type_id = dt.type_id
inner join customer c on d.customer_id = c.customer_id
group by d.Deliver_id;
```

-- 12 --

```
select * from delivery d
inner join Delivery_Detail dd on d.Deliver_id = dd.Deliver_id
inner join Delivery_Type dt on d.Type_id = dt.type_id
inner join customer c on d.customer_id = c.customer_id
where d.Deal_id = c.First_Name = "Anar" and c.Last_Name = "Djafarov";
```

-- 13 --

```
select d.Deliver_id, d.Status_Delivery from delivery d
inner join Delivery_Type dt on d.Type_id = dt.type_id
inner join customer c on d.customer_id = c.customer_id
where d.Deliver_id = 1;
```

-- 14 --

```
select sum(Amount * dd.Weight * dt.Cost) as Delivery_Price , dt.Type_Delivery
from delivery d
inner join Delivery_Detail dd on d.Deliver_id = dd.Deliver_id
inner join Delivery_Type dt on d.Type_id = dt.type_id
inner join customer c on d.customer_id = c.customer_id
where d.Start_Date >= "2020-02-01" AND d.Start_Date <= "2020-03-01"
and dt.Type_Delivery = "Courier_Company xpress"
group by dt.Type_Delivery;
```

-- 15 --

```
select Sum(d.Total_Cost) as Total_Cost , d.Pay_Option from Deal d
where d.Pay_Option = "bit"
and d.Deal_Date >= "2020-07-01" AND d.Deal_Date <= "2020-08-01"
group by d.Pay_Option;
```

-- 16 --

```

SELECT d.deal_id, sum((dd.Sell_Price - pd.price_per_one) * dd.Amount) AS
book_profit FROM Deal d
inner join Deal_Details dd on d.Deal_id = dd.Deal_id
inner join inventory i on dd.Copy_id = i.Copy_id
inner join Purchasing_Detail pd on dd.Copy_id = pd.Copy_id
where d.Deal_Date >= 12-MONTH(NOW()) AND d.Deal_Date <= NOW()
GROUP BY d.deal_id
HAVING (
    SELECT AVG(Profit) AS average FROM
        (select sum((dd.Sell_Price - pd.price_per_one) * dd.Amount)
        as Profit from Deal d
        inner join Deal_Details dd on d.Deal_id = dd.Deal_id
        inner join inventory i on dd.Copy_id = i.Copy_id
        inner join Purchasing_Detail pd on dd.Copy_id =
        pd.Copy_id
        inner join purchasing p on p.Purchase_id = pd.Purchase_id
        where d.Deal_Date >= 12-MONTH(NOW()) AND
        d.Deal_Date <= NOW()
        group by d.deal_id) AS Average_Profit
    ) < book_profit ;

```

-- 17 --

```

select count(d.Deliver_id) Num_Of_Delivers, dt.Type_Delivery from delivery d
inner join Delivery_Detail dd on d.Deliver_id = dd.Deliver_id
inner join Delivery_Type dt on d.Type_id = dt.type_id
where d.Start_Date >= 12-MONTH(NOW()) AND d.Start_Date <= NOW()
group by dt.Type_Delivery;

```

-- 18 --

```
select count(distinct(b.book_id)) Num_Of_Copys , d.Deliver_id, b.Book_Name ,
bc.Copy_id from delivery d
inner join Delivery_Detail dd on d.Deliver_id = dd.Deliver_id
inner join Delivery_Type dt on d.Type_id = dt.type_id
inner join Book_Copy BC on dd.Copy_id = BC.Copy_id
inner join book b on b.Book_id = BC.Book_id
group by d.Deliver_id having Num_Of_Copys > 1
```

-- 19 --

```
select * from Customer;
```

-- 20 --

```
select SO.Order_id, SO.Amount, SO.Order_Date, SO.Order_Status,
CONCAT(c.First_Name, " ", c.Last_Name ) as Customer_Name from
Store_Order SO
inner join Book_Copy BC on SO.Copy_id = BC.Copy_id
inner join book b on b.Book_id = BC.Book_id
inner join customer c on SO.customer_id = c.customer_id
where SO.Order_Status = "in_Storage"
and SO.Order_Date >= 14-Day(NOW()) AND SO.Order_Date <= NOW();
```

-- 21 --

```
select Sum(i.New_Amount) as Amount, YEAR(i.Stored_Date) AS 'year',
MONTH(i.Stored_Date) AS 'month' from inventory i
where i.Stored_Location = "Stock"
and i.Stored_Date >= "2020-07-01" AND i.Stored_Date <= "2020-08-01";
```

-- 22 - A --

```
select Sum(pd.Amount) Books_Total_Amount, Sum(pd.Amount *
pd.Price_Per_One) Books_Total_Price from purchasing p
inner join Purchasing_Detail pd on p.Purchase_id = pd.Purchase_id
where p.Purchasing_Date >= "2020-07-01" AND p.Purchasing_Date <= "2020-08-01";
```

-- 22- B--

```
select sum((dd.Sell_Price - pd.price_per_one) * dd.Amount) as Profit ,
YEAR(d.Deal_Date) AS 'year', MONTH(d.Deal_Date) AS 'month' from Deal d
inner join Deal_Details dd on d.Deal_id = dd.Deal_id
inner join inventory i on dd.Copy_id = i.Copy_id
inner join Purchasing_Detail pd on dd.Copy_id = pd.Copy_id
inner join purchasing p on p.Purchase_id = pd.Purchase_id
where d.Deal_Date >= "2020-07-01" AND d.Deal_Date <= "2020-08-01";
```

-- 23 --

```
select avg(d.Total_Cost) Total_Year_Cost, Month(d.Deal_Date) as 'Month',
Year(d.Deal_Date)-1 as 'From_Year', Year(d.Deal_Date) as 'To_Year' from Deal
d
where d.Deal_Date >= "2019-08-01" AND d.Deal_Date <= "2020-08-01"
GROUP BY MONTH(d.Deal_Date);
```

-- 24 --

```
select CONCAT(W.First_Name, " ", W.Last_Name ) as Worker_Name,
(W.Hourly_Salary * SW.Hours) as Bruto_Salary, SW.Month_Val As Month,
SW.Year_Val As Year from store_worker W
inner join Summery_Worker SW on w.Worker_id = SW.Worker_id
where W.First_Name = "Jhonatan" and W.Last_Name ="Cubani"
and SW.Month_Val = 2 and SW.Year_Val = 2020 ;
```

-- 25 --

```
select count(d.deal_id) Deal_Number, CONCAT(SW.First_Name, " ",
SW.Last_Name ) as Worker_Name from Deal d
inner join store_worker SW on SW.Worker_id = d.Worker_id
where d.Deal_Date >= "2020-07-01" AND d.Deal_Date <= "2020-08-01"
group by SW.Worker_id limit 1;
```

8. מסמך אלגברת יחסים



1.  $\pi$  Inventory ((Book\_Name  $\bowtie$  New\_Amount as Amount)  $\bowtie$  Stored\_Location) ((Inventory  $\bowtie$  Book\_Copy BC on = Copy\_id)  $\bowtie$  book b on Book\_id = BC.Book\_id) ( $\sigma$  Book\_Name = "X") p (i.Sequence))

2.  $\pi$  coustumer ( $\sigma$  Customer order by Sign\_Up\_Date limit 1 )

3.  $\pi$  Inventory ((Copy\_id  $\bowtie$  Stored\_Date)  $\bowtie$  Book\_Name) ( (Inventory  $\bowtie$  Book\_Copy BC on = Book\_id = i.Copy\_id )  $\bowtie$  (book on Book\_id = BC.Book\_id)) order by i.Stored\_Date  
limit 1;

4.  $\pi$  Store\_Order ( $\sigma$  (Store\_Order Status != "Complete"  
order by SO.Order\_Date)

5.  $\pi$  Deal (((U (dd.Amount) as Num\_Of\_Books )  $\bowtie$  (Deal\_Details dd on d.Deal\_id = dd.Deal\_id))  $\bowtie$  ((Book\_Copy BC on BC.Book\_id = dd.Copy\_id)  $\bowtie$  (b on b.Book\_id = BC.Book\_id(  $\sigma$  b.Book\_Name = "x" (p( dd.Amount)))

6.  $\pi$  Deal (((U (dd.Amount) as (Selling\_Books  $\bowtie$  Author\_Name) )  $\bowtie$  (Deal\_Details dd on d.Deal\_id = dd.Deal\_id  
)  $\bowtie$  ( Book\_Copy BC on dd.Copy\_id = BC.Copy\_id))  $\bowtie$  (( book b on b.Book\_id = BC.Book\_id)  $\bowtie$  ( author a on a.Author\_id = BC.Author\_id))  
( $\sigma$  (Deal\_Date >= x ^ Deal\_Date <= y)( p(A.Author\_Name)  $\bowtie$  ( ORDER BY ^ (dd.Amount) DESC  
LIMIT 1;

7.  $\pi$  Deal (((U (dd.Amount) as (Buing\_Books  $\bowtie$  CONCAT(c.First\_Name, " ", c.Last\_Name ) as Customer\_Name ) )  $\bowtie$  (Deal\_Details dd on d.Deal\_id = dd.Deal\_id)  $\bowtie$  ( Book\_Copy BC on dd.Copy\_id = BC.Copy\_id))  $\bowtie$  (( book b on b.Book\_id = BC.Book\_id)  $\bowtie$  ( customer c on d.customer\_id = c.customer\_id))  
p(d.Customer\_id)  $\bowtie$  ( ORDER BY ^ (d.customer\_id) DESC  
LIMIT 3;

8.  $\pi$  Inventory (COUNT(distinct(BC.Translation\_id) ((U Translation\_Num U b.Book\_Name) ))  $\bowtie$  ((Book\_Copy BC on i.Copy\_id = BC.Copy\_id)  $\bowtie$  (book b on b.Book\_id = BC.Book\_id)  $\bowtie$  (t on t.Translation\_id = BC.Translation\_id))  
(p(b.book\_id) order by count(distinct(BC.Translation\_id)) DESC  
limit 1;

9.  $\pi$  Deal(((U CONCAT(c.First\_Name, " ", c.Last\_Name ) as Customer\_Name) U (b.book\_name) U (dd.sell\_price) U (d.Deal\_Date ))  $\bowtie$  (Deal\_Details dd on d.Deal\_id = dd.Deal\_id)  $\bowtie$  (Book\_Copy BC on dd.Copy\_id = BC.Copy\_id)  $\bowtie$  (book b on b.Book\_id = BC.Book\_id)  $\bowtie$  (customer c on d.customer\_id = c.customer\_id)  $\bowtie$  ( $\sigma$ (c.First\_Name = "X" U c.Last\_Name = "Y") order by d.Deal\_Date)

10.  $\pi$  Store\_Order (((U CONCAT(c.First\_Name, " ", c.Last\_Name ) as Customer\_Name) U (SO.Order\_Date) U (b.Book\_Name) U (b.Title) U (b.Book\_Condition) U (SO.Amount) U (So.Order\_Status))  $\bowtie$  (Book\_Copy BC on dd.Copy\_id = BC.Copy\_id)  $\bowtie$  (book b on b.Book\_id = BC.Book\_id)  $\bowtie$  (customer c on d.customer\_id = c.customer\_id)  $\bowtie$  ( $\sigma$ (c.First\_Name = "X" U c.Last\_Name = "Y") order by SO.Order\_Date)

11.  $\pi$  Delivery ((( $\cup$  CONCAT( $c$ .First\_Name, " ",  $c$ .Last\_Name ) as Customer\_Name)  $\cup$  (sum(Amount \*  $dd$ .Weight \*  $dt$ .Cost) as Delivery\_Price)  $\bowtie$  (Delivery\_Detail  $dd$  on  $d$ .Deliver\_id =  $dd$ .Deliver\_id)  $\bowtie$  (Delivery\_Type  $dt$  on  $d$ .Type\_id =  $dt$ .type\_id)  $\bowtie$  (customer  $c$  on  $d$ .customer\_id =  $c$ .customer\_id)  $\bowtie$  ( $\sigma$ ( $c$ .First\_Name = "X"  $\cup$   $c$ .Last\_Name = "Y") order by SO.Order\_Date)

12.  $\pi$  delivery ((Delivery\_Detail  $dd$  on  $d$ .Deliver\_id =  $dd$ .Deliver\_id  $\bowtie$  (Delivery\_Type  $dt$  on  $d$ .Type\_id =  $dt$ .type\_id)  $\bowtie$  (customer  $c$  on  $d$ .customer\_id =  $c$ .customer\_id)  $\bowtie$   $\bowtie$  ( $\sigma$ ( $c$ .First\_Name = "X"( $c$ .First\_Name = "Y"  $\cup$   $c$ .Last\_Name = "Z"))))

13.  $\pi$  Delivery ((( $d$ .Deliver\_id)  $\cup$  ( $d$ .Status\_Delivery))  $\bowtie$  (Delivery\_Type  $dt$  on  $d$ .Type\_id =  $dt$ .type\_id)  $\bowtie$  (customer  $c$  on  $d$ .customer\_id =  $c$ .customer\_id)  $\bowtie$  ( $\sigma$ ( $d$ .Deliver\_id = 1))

14.  $\pi$  Delivery ((( $\cup$  ( $dt$ .Type\_Delivery )  $\cup$  (sum(Amount \*  $dd$ .Weight \*  $dt$ .Cost) as Delivery\_Price)  $\bowtie$  (Delivery\_Detail  $dd$  on  $d$ .Deliver\_id =  $dd$ .Deliver\_id)  $\bowtie$  (Delivery\_Type  $dt$  on  $d$ .Type\_id =  $dt$ .type\_id)  $\bowtie$  (customer  $c$  on  $d$ .customer\_id =  $c$ .customer\_id)  $\bowtie$  ( $\sigma$ ( $d$ .Start\_Date  $\geq$  "X"  $\cup$   $d$ .Start\_Date  $\leq$  "Y") $\cup$  (( $dt$ .Type\_Delivery = "Courier\_Company xpress")  $p$ ( $dt$ .Type\_Delivery))

15.  $\pi$  Deal((( $\cup$  (sum( $d$ .Total\_Cost) as Total\_Cost) ( $\cup$  ( $d$ .Pay\_Option)) ( $\sigma$ ( $d$ .Pay\_Option = "bit") $\cup$  ( $d$ .Deal\_Date  $\geq$  "X") $\cup$  ( $d$ .Deal\_Date  $\leq$  "Y")  $p$ ( $d$ .Pay\_Option)

17.  $\pi$  Delivery ((( $\cup$  (d.Deliver\_id) Num\_Of\_Delivers)  $\cup$  (dt.Type\_Delivery )  $\bowtie$   
 (Delivery\_Detail dd on d.Deliver\_id = dd.Deliver\_id)  $\bowtie$   
 (Delivery\_Type dt on d.Type\_id = dt.type\_id)  $\bowtie$   
 ( $\sigma$ (d.Start\_Date  $\geq$  12-MONTH(NOW())  $\cup$  d.Start\_Date  $\leq$  NOW()))  
 p(dt.Type\_Delivery))

18.  $\pi$  Delivery ((( $\cup$  count(distinct(b.book\_id)) Num\_Of\_Copys)  $\cup$  (d.Deliver\_id) $\cup$   
 (b.Book\_Name) $\cup$  (bc.Copy\_id))  $\bowtie$   
 (Delivery\_Detail dd on d.Deliver\_id = dd.Deliver\_id)  $\bowtie$   
 (Delivery\_Type dt on d.Type\_id = dt.type\_id)  $\bowtie$   
 (Book\_Copy BC on dd.Copy\_id = BC.Copy\_id)  $\bowtie$   
 (book b on b.Book\_id = BC.Book\_id)  $\bowtie$   
 (p(d.Deliver\_id having Num\_Of\_Copys > 1))

19.  $\pi$  Customer

20.  $\pi$  Store\_Order ((( $\cup$  CONCAT(c.First\_Name, " ", c.Last\_Name ) as  
 Customer\_Name)  $\cup$  (SO.Order\_id)  $\cup$  (SO.Amount)  $\cup$  (SO.Order\_Date)  $\cup$   
 (SO.Order\_Status) $\bowtie$   
 (Book\_Copy BC on dd.Copy\_id = BC.Copy\_id)  $\bowtie$   
 (book b on b.Book\_id = BC.Book\_id)  $\bowtie$   
 (customer c on d.customer\_id = c.customer\_id)  $\bowtie$   
 ( $\sigma$ (SO.Order\_Status = in\_Storage)  $\cup$  ( SO.Order\_Date  $\geq$  14-Day(NOW()) ) $\cup$   
 ( SO.Order\_Date  $\leq$  NOW()))

21.  $\pi$  Inventory ((Sum(i.New\_Amount) as Amount)  $\cup$  (YEAR(i.Stored\_Date) AS  
 'year')  $\cup$   
 (MONTH(i.Stored\_Date) AS 'month')  $\bowtie$   
 ((Book\_Copy BC on i.Copy\_id = BC.Copy\_id)  $\bowtie$   
 ( $\sigma$ (i.Stored\_Location = "Stock")  $\cup$  ( i.Stored\_Date  $\geq$  "X" ) $\cup$   
 ( i.Stored\_Date  $\leq$  "Y"))

22A.  $\pi$  Purchasing ((Sum(pd.Amount) Books\_Total\_Amount)  $\cup$   
 (Sum(pd.Amount \* pd.Price\_Per\_One Books\_Total\_Price)  $\bowtie$   
 ((Purchasing\_Detail pd on p.Purchase\_id = pd.Purchase\_id)  $\bowtie$   
 ( $\sigma$ (p. Purchasing\_Date  $\geq$  "X")  $\cup$  ( p.Purchasing\_Date  $\leq$  "Y" )

22b.  $\pi$  Deal((( $\cup$  (sum(dd.Sell\_Price - pd.price\_per\_one) \* dd.Amount) as Profit)  $\cup$   
 (YEAR(d.Deal\_Date) AS 'year')  $\cup$  ( MONTH(d.Deal\_Date) AS 'month')  $\bowtie$   
 ((Deal\_Details dd on d.Deal\_id = dd.Deal\_id)  $\bowtie$

(inventory i on dd.Copy\_id = i.Copy\_id)  $\bowtie$   
 (Purchasing\_Detail pd on dd.Copy\_id = pd.Copy\_id)  $\bowtie$   
 (purchasing p on p.Purchase\_id = pd.Purchase\_id)  $\bowtie$   
 ( $\sigma$ ( d.Deal\_Date  $\geq$  "X") $\cup$ ( d.Deal\_Date  $\leq$  "Y")

23.  $\pi$  Delivery ((avg(d.Total\_Cost) Total\_Year\_Cost)  $\cup$  (Month(d.Deal\_Date) as  
 'Month')  $\cup$   
 (Year(d.Deal\_Date)-1 as 'From\_Year')  $\cup$  ( Year(d.Deal\_Date) as 'To\_Year'))  $\bowtie$   
 ( $\sigma$ (d.Deal\_Date  $\geq$  "X")  $\cup$  (d.Start\_Date  $\leq$  "Y")  
 p(MONTH(d.Deal\_Date))

24.  $\pi$  store\_worker ((( $\cup$  CONCAT(c.First\_Name, " ", c.Last\_Name ) as  
 Worker\_Name)  $\cup$  (W.Hourly\_Salary \* SW.Hours) as Bruto\_Salary)  $\cup$   
 (SW.Month\_Val As Month)  $\cup$  (SW.Year\_Val As Year))  $\bowtie$   
 (Summery\_Worker SW on w.Worker\_id = SW.Worker\_id)  $\bowtie$   
 ( $\sigma$ (W.First\_Name = "X")  $\cup$  ( W.Last\_Name ="Y") $\cup$   
 ( SW.Month\_Val = 2)  $\cup$  SW.Year\_Val = 2020))

25.  $\pi$  Deal ((( $\cup$  CONCAT(SW.First\_Name, " ", SW.Last\_Name ) as  
 Worker\_Name)  $\cup$  (count(d.deal\_id) Deal\_Number))  $\bowtie$   
 (store\_worker SW on SW.Worker\_id = d.Worker\_id)  $\bowtie$   
 ( $\sigma$ (d.Deal\_Date  $\geq$  "X"  $\cup$  d.Deal\_Date  $\leq$  "Y") p(SW.Worker\_id  
 limit 1))