

פרויקט תכנות

קורס – 'מערכות קבצים ומסדי נתונים'

מרצה – יצחק נודלר

מגישים –

גיא שריר 310010244

אופאל פלצמן 208521385

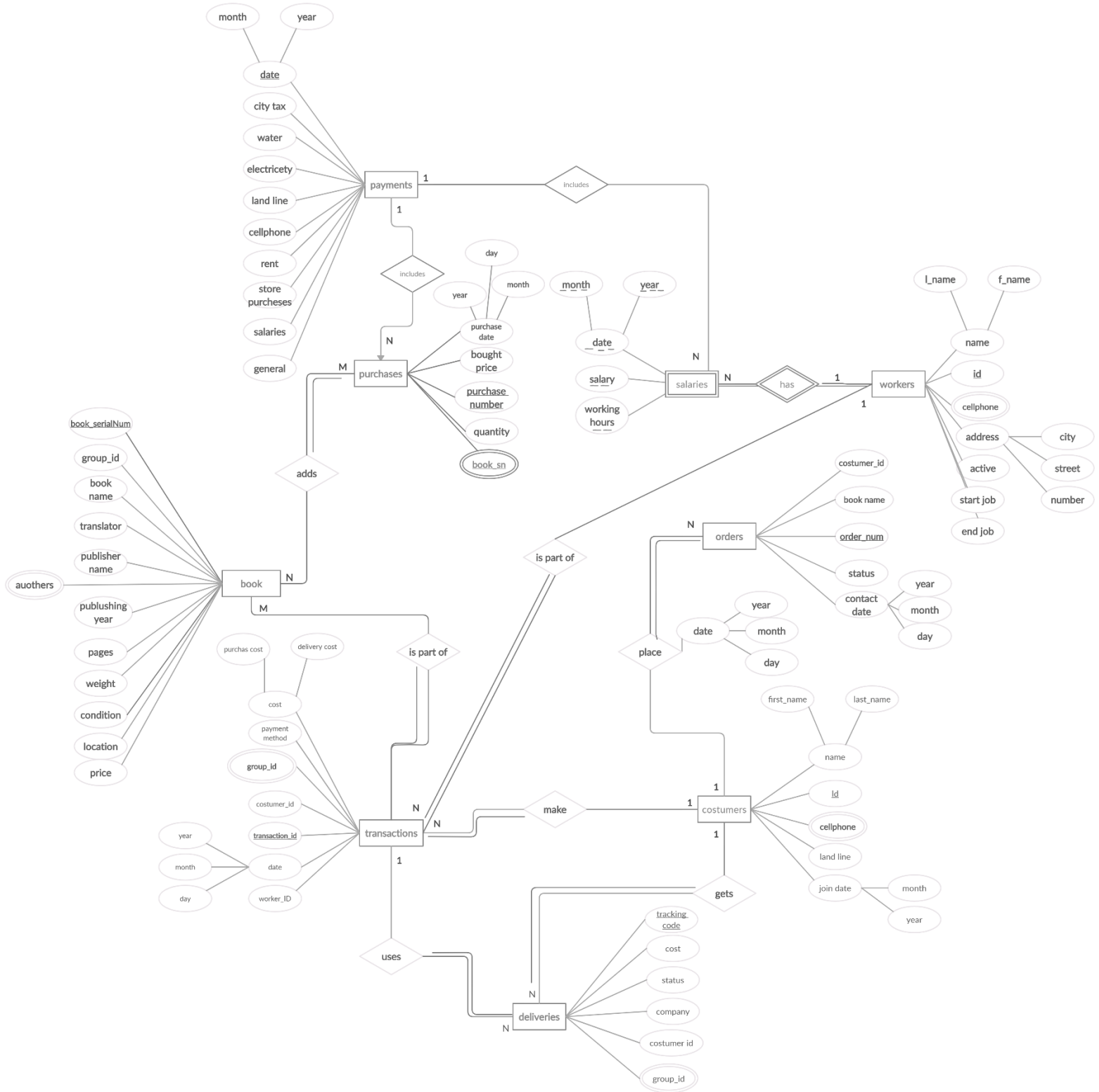
הערה חשובה-

באפליקציה עבור ממשק מנהל החנות

יש להזין סיסמא.

הסיסמא- sqlforlife

## 1. מידול מסד הנתונים במודל ה:ER



מקרא סימונים בסעיף  
ג.1

## 5.2 מסמכים נלווים

### הערות למודל ER

1.א. ציון המפתחות

Primary key for each table:

<u>מפתחות</u>	<u>ישות/קשר</u>
Worker_id	Workers
Worker_id, date_m, date_y	Salaries
purchase_number	Purchases
book_sn	purchases_books(adds)
book_sn	Book
tracking_code	Deliveries
Transaction_id	Transactions
order_num	Orders
Transaction_id, group_id	transaction_group_id
date_m, date_y	Payments
costumer_id	Costumers
order_num	Costumer_orders
Cellphone	Cosumers_cellphone
Cellphone	workers_cellphone
book_sn, author	book_authors
group_id	book_group_id
tracking_code, group_id	deliveries_group_id
Transaction_id, book_sn	transaction_books(is part of)

## 1.ב. הסברים על הדיאגרמה:

1.

ישות	קשר	ישות
Salary	has	workers
השתתפות מלאה	-	השתתפות מלאה
N	-	1

Salary היא ישות חלשה = weak entity, אין לה מפתח ראשי.

לכן במעבר ממודל ER לסכמה ראשונית הישות salary תקבל את המפתח הראשי של הישות החזקה שיש ביניהן קשר. בדוגמא שלנו תקבל את המפתח worker\_id של ישות workers.

עבור ישות חלשה salaries שאר תכונותיה מהוות partial key.

### השתתפות מלאה/חלקית

השתתפות salary בקשר 'has' היא השתתפות מלאה מכיוון שכל salary חייב להיות קשור בקשר has עם worker.

השתתפות worker בקשר 'has' היא השתתפות מלאה מכיוון שכל worker חייב להיות קשור בקשר has עם salary. (תחת ההנחה שעבור העולם אותו אנו ממדלים ייצוג worker מתאפשר רק עבור worker שיש לו salary אחד לפחות).

### אילוץ היחס

לכל worker יכולות להיות מספר רב של salaries.

לכל תלוש משכורת, salary שייכות ל worker ספציפי.

2.

ישות	קשר	ישות
order	place	costumers
השתתפות מלאה	-	השתתפות חלקית
N	-	1

### השתתפות מלאה/חלקית

השתתפות order בקשר 'place' היא השתתפות מלאה מכיוון שכל order חייב להיות קשור בקשר place עם costumer. (הזמנה מתבצעת ע"י לקוח שמחפש ספר מסוים).

השתתפות costumer בקשר 'place' היא השתתפות חלקית מכיוון שלא כל costumer חייב להיות קשור בקשר place עם order. (לפי הגדרת העולם, לקוח הינו כל אדם שמסר את פרטיו והם נרשמו במסד הנתונים)

### אילוץ היחס

לכל costumer יכולות להיות מספר רב של orders.

לכל הזמנה, שייכות ל costumers ספציפי.

#### הערות

מכיוון שאילוף היחס הוא 1:N במעבר לסכמה ראשונית נוסף לקבוצת הישויות order את ה primary key של קבוצת הישויות costumers .

.3

ישות	קשר	ישות
transactions	make	costumers
השתתפות מלאה	-	השתתפות חלקית
N	-	1

#### השתתפות מלאה/חלקית

השתתפות transactions בקשר 'make' היא השתתפות מלאה מכיוון שכל transactions חייב להיות קשור בקשר make עם costumers . (עסקה מתבצעת ע"י לקוח).

השתתפות costumers בקשר 'make' היא השתתפות חלקית מכיוון שלא כל costumers חייב להיות קשור בקשר make עם transactions . (לפי הגדרת העולם, לקוח הינו כל אדם שמסר את פרטיו והם נרשמו במסד הנתונים)

#### אילוף היחס

לכל costumers יכול להיות מספר רב של transactions .

לכל עסקה, transactions שייכות ל costumers ספציפי.

#### הערות

מכיוון שאילוף היחס הוא 1:N במעבר לסכמה ראשונית נוסף לקבוצת הישויות transactions את ה primary key של קבוצת הישויות costumers .

.4

ישות	קשר	ישות
transactions	uses	deliveries
השתתפות חלקית	-	השתתפות מלאה
1	-	N

#### השתתפות מלאה/חלקית

השתתפות deliveries בקשר 'uses' היא השתתפות מלאה מכיוון שכל deliveries חייב להיות קשור בקשר uses עם transactions . (משלוח יכול להתבצע בעת ביצוע עסקה).

השתתפות transactions בקשר 'uses' היא השתתפות חלקית מכיוון שלא כל transactions חייב להיות קשור בקשר uses עם deliveries . (בתיאור העולם רכישת ספר יכולה להתבצע גם על ידי הגעה לחנות ורכישה)

### אילוץ היחס

לכל עסקה, transaction יכול להיות מספר רב של deliveries.

לכל משלוח, delivery שייכות ל transaction ספציפי.

### הערות

מכיוון שאילוץ היחס הוא 1:N במעבר לסכמה ראשונית נוסף לקבוצת הישויות deliveries את ה primary key של קבוצת הישויות transactions .

5.

ישות	קשר	ישות
transactions	Is part of	book
השתתפות מלאה	-	השתתפות חלקית
N	-	M

### השתתפות מלאה/חלקית

השתתפות transactions בקשר 'Is part of' היא השתתפות מלאה מכיוון שכל transactions חייב להיות קשור בקשר Is part of עם book . (בתיאור העולם, 'רכישת לקוח' – קניה של ספר מהחנות).

השתתפות book בקשר 'Is part of' היא השתתפות חלקית מכיוון שלא כל book חייב להיות קשור בקשר Is part of עם transactions . (לא כל ספר חייב "להשתתף" בעסקה)

### אילוץ היחס

לכל עסקה, transaction יכול להיות מספר רב של books.

לכל ספר, שייכות ל transaction ספציפי.

### הערות

מכיוון שלכל עסקה יכול להיות מספר רב של ספרים, זוהי תכונה מרובת ערכים עבור ישות 'עסקה', transaction. לכן במעבר לסכמה ראשונית נייצר טבלה נוספת שתכיל עבור כל מספר עסקה ספר השייך לאותה עסקה.

עבור קשר M:N נייצר טבלה בסכמה ראשונית. טבלה זו תכיל את המפתח של טבלת book ואת המפתח של טבלת transactions.

6.

ישות	קשר	ישות
costumer	gets	deliveries
השתתפות חלקית	-	השתתפות מלאה
1	-	N

#### השתתפות מלאה/חלקית

השתתפות deliveries בקשר 'gets' היא השתתפות מלאה מכיוון שכל deliveries חייב להיות קשור בקשר gets עם costumer. (בתיאור העולם משלוח מתבצע עבור לקוח).

השתתפות costumer בקשר 'gets' היא השתתפות חלקית מכיוון שלא כל costumer חייב להיות קשור בקשר gets עם deliveries. (לא כל לקוח חייב להשתמש בשירותי משלוח)

#### אילוץ היחס

לכל לקוח, costumer יכול להיות מספר רב של deliveries.

לכל משלוח, delivery שייכות ל costumer ספציפי.

#### הערות

מכיוון שאילוץ היחס הוא 1:N במעבר לסכמה ראשונית נוסיף לקבוצת הישויות deliveries את ה primary key של קבוצת הישויות costumer.

7.

ישות	קשר	ישות
transactions	Is part of	worker
השתתפות מלאה	-	השתתפות חלקית
N	-	1

#### השתתפות מלאה/חלקית

השתתפות transactions בקשר 'Is part of' היא השתתפות מלאה מכיוון שכל transactions חייב להיות קשור בקשר Is part of עם worker. (ביצוע עסקה מחייבת עובד שיבצע ויזין את נתוני העסקה).

השתתפות worker בקשר 'Is part of' היא השתתפות חלקית מכיוון שלא כל worker חייב להיות קשור בקשר Is part of עם transactions. (ההנחה עבור העולם אותו אנו ממדלים היא שלא כל עובד חייב לבצע עסקה)

#### אילוץ היחס

לכל עובד, worker יכול להיות מספר רב של transactions.

לכל עסקה, transactions שייכות ל worker ספציפי.

## הערות

מכיוון שאילוץ היחס הוא 1:N במעבר לסכמה ראשונית נוסיף לקבוצת הישויות transactions את ה primary key של קבוצת הישויות worker .

8.

ישות	קשר	ישות
book	adds	purchases
השתתפות חלקית	-	השתתפות מלאה
N	-	M

## השתתפות מלאה/חלקית

השתתפות purchases בקשר 'adds' היא השתתפות מלאה מכיוון שכל purchases חייב להיות קשור בקשר adds עם book . (בתיאור העולם, 'רכישת חנות' – קנייה של ספר על ידי החנות).

השתתפות book בקשר 'adds' היא השתתפות חלקית מכיוון שלא כל book חייב להיות קשור בקשר adds עם purchases . (ההנחה היא שספר יכול להימצא בעולם אותו אנו ממדלים ללא קשר לרכישה)

## אילוץ היחס

בכל רכישה ניתן לרכוש מספר רב של ספרים.

ספר בעל מספר סידורי X יכול להופיע ברכישה אחת .

## הערות

מכיוון שבקניה, ניתן לרכוש מספר ספרים, התכונה book\_sn היא תכונה מרובת ערכים עבור ישות purchases. לאור עובדה זו נייצר טבלה נוספת במעבר לסכמה ראשונית שתכיל עמודה book\_sn ועמודה purchase\_num.

עבור קשר M:N נייצר טבלה בסכמה ראשונית. טבלה זו תכיל את המפתח של טבלת book ואת המפתח של טבלת purchases.

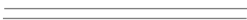


1.g. מקרא סימונים עבור תרשימים ER:

הסבר

סימון

השתתפות מלאה בקשר



השתתפות חלקית בקשר



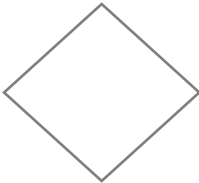
Strong entity



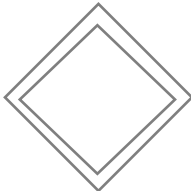
Weak entity



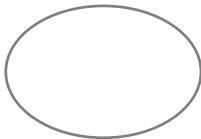
Relationship



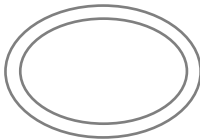
Weak relationship



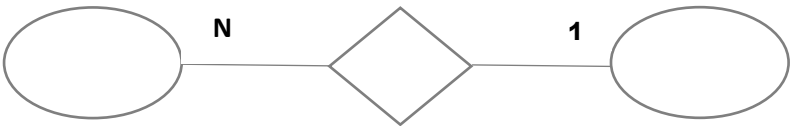
Attribute



Multivalued attribute



One: N on



## 2. הנחות ליצירת תרשים ה-ERD:

1. תאריך משלוח הינו תאריך יצירת העסקה דרכה נעשה השימוש בשירותי משלוח.
2. הספרים מחולקים למהדורות כך שבכל מהדורה שוני באחד או יותר מהערכים הנ"ל:  
Translator, publisher\_name, publishing\_year.  
בכל מהדורה הערכים לעיל חד חד ערכיים.
3. ללקוח מספר טלפון נייד אחד או יותר.
4. עבור orders, סטטוס 'waiting' משמעו שהספר עדיין לא נרכש/הגיע לחנות לכן גורר contact date שערכו NULL.
- סטטוס 'updated' משמעו שהספר נמצא בחנות ונוצר קשר עם הלקוח על מנת ליידע אותו, ולכן contact date יכיל תאריך יצירת הקשר של החנות עם הלקוח.
5. רווח החנות בחודש מסוים כפי שצוין ב'הערות' בשאילתה 22 בסעיף 3.6 הינו סך סכומי רכישת-לקוח פחות סך סכומי רכישת-חנות.
6. עסקה מתבצעת על ידי עובד יחיד.
7. הזמנת ספר מתייחסת למהדורה של הספר ולא למספר סידורי, כלומר לקוח יכול לרכוש ספר ממהדורה X, אך לא לרכוש מספר סידורי. מספר הסידורי של הספר אותו רכש מחושב אוטומטית מהספרים הזחים הנמצאים במהדורה X.
8. כאשר ספר נרכש על ידי לקוח, הספר נמחק מטבלת books, כלומר המספר הסידורי של אותו ספר נמחק מהמלאי של החנות, אך המהדורה אליה הספר שייך קיימת.
9. מחירי משלוח של חברות השליחויות השונות אינו קבוע ולכן אינו נקבע בבסיס הנתונים אלא מעודכן באפליקציה.
10. לא קיים ספר שלא תורגם.
11. מחירי הספרים, עלויות משלוחים, תשלומים ועוד נספרים במטבע שקל חדש.
12. עסקה ניתן לבצע על ידי אמצעי תשלום יחיד.
13. הזמנת לקוח מתייחסת לשם ספר ולא למהדורה.
14. מספר טלפון קווי עבור לקוח יכול להיות NULL.
15. עבור עובד שעדיין מועסק על ידי החנות זמן סיום העבודה יהיה NULL.

### 3. תלויות פונקציונליות מהעולם (F):

תלויות פונקציונליות	ישות
<ul style="list-style-type: none"> <li>· group_id -&gt; author, translator, name, publish_year, publish_name, weight, pages</li> <li>· Book_sn -&gt; location, group_id, book_name, book_price, translator, publish_name, authors, publish_year, pages, weight, condition</li> </ul>	<u>Book</u>
<ul style="list-style-type: none"> <li>· Transaction_id -&gt; costumer_id, trans_cost, delivery_cost, pay_method, year, month, day, worker_id, book_group_id</li> </ul>	<u>Transactions</u>
<ul style="list-style-type: none"> <li>· Tracking_code -&gt; cost, status, company, del_year, del_month, del_day, costumer_id, transaction_id</li> </ul>	<u>Deliveries</u>
<ul style="list-style-type: none"> <li>· purch_num -&gt; bought_price, date_y, date_m, date_d, quantity, book_sn</li> </ul>	<u>Purchases</u>
<ul style="list-style-type: none"> <li>worker_id, month, year -&gt; salary, working_hours</li> </ul>	<u>Salaries</u>
<ul style="list-style-type: none"> <li>· costumer_id -&gt; f_name, l_name, join_date, landline</li> <li>· cellphone -&gt; f_name, l_name, join_date, landline, costumer_id</li> </ul>	<u>Costumers</u>
<ul style="list-style-type: none"> <li>· order_num -&gt; costumer_id, book_name, status, contact_date</li> </ul>	<u>Orders</u>
<ul style="list-style-type: none"> <li>· month, year -&gt; city_tax, water, electricity, landline, cellular, rent, store_purchases, salaries, general</li> </ul>	<u>Payments</u>
<ul style="list-style-type: none"> <li>· Worker_id -&gt; l_name, f_name, city, street, house_num, start_job, end_job</li> </ul>	<u>Workers</u>

#### 4. סכמה ראשונית של מודל ה ER:

##### Book-

<u>Book_sn</u>	Book_name	translator	Publ_name	Publ_year	pages	weigh	condition	group_id	location	price
----------------	-----------	------------	-----------	-----------	-------	-------	-----------	----------	----------	-------

##### Book\_author-

<u>Book_sn</u>	<u>author</u>
----------------	---------------

##### Transactions-

trans_cost	Delivery_cost	Pay_method	Costumer_id	<u>Transaction_id</u>	year	month	day	Worker_id
------------	---------------	------------	-------------	-----------------------	------	-------	-----	-----------

##### Transactions book-

<u>Transaction_id</u>	<u>Book_sn</u>
-----------------------	----------------

##### Deliveries-

<u>Track_code</u>	cost	status	company	Costumer_id	year	month	day	Transaction_id
-------------------	------	--------	---------	-------------	------	-------	-----	----------------

##### Delivery items

<u>Tracking_code</u>	<u>Book_sn</u>
----------------------	----------------

##### Costumers cellphone-

<u>Costumer_id</u>	<u>cellphone</u>
--------------------	------------------

##### Costumer-

f_name	L_name	<u>id</u>	landline	Join_year	Join_month
--------	--------	-----------	----------	-----------	------------

##### Orders-

Costumer_id	Book_name	<u>Order_num</u>	status	Contact_year	Contact_month	Contact_day
-------------	-----------	------------------	--------	--------------	---------------	-------------

##### Customer\_orders

Order_y	Order_m	Order_d	Costumer_id	<u>Order_num</u>
---------	---------	---------	-------------	------------------

**Purchases-**

<u>year</u>	<u>month</u>	<u>day</u>	Bought_price	<u>Purch_num</u>	quantity
-------------	--------------	------------	--------------	------------------	----------

**Purchases\_book**

<u>Purch_num</u>	<u>Book_sn</u>
------------------	----------------

**Payments-**

<u>year</u>	<u>month</u>	City_tx	water	electri	landline	cellular	rent	Store_purch	salaries	general
-------------	--------------	---------	-------	---------	----------	----------	------	-------------	----------	---------

**Workers-**

<u>l_name</u>	<u>f_name</u>	<u>id</u>	city	street	House_number	Start_date	End_date
---------------	---------------	-----------	------	--------	--------------	------------	----------

**Workers\_cellphone-**

<u>Worker_id</u>	<u>cellphone</u>
------------------	------------------

**Salaries-**

<u>year</u>	<u>month</u>	salary	Working_hours	<u>Worker_id</u>
-------------	--------------	--------	---------------	------------------

**Location\_quantity-**

<u>location</u>	<u>date</u>	<u>quantity</u>
-----------------	-------------	-----------------

**תהליך נירמול- מציאת כל הת"פ (הסגור של F)**

**נראה את הת"פ הדורשות שיפור בטבלאות:**

**Book-**

1. Book\_sn -> location, group\_id, book\_name, book\_price, translator, publish\_name, authors, publish\_year, pages, weight, condition

2.group\_id -> translator, book\_name, publish\_year, publish\_name, pages, weight

**מצאנו שקיימת תלות פונקציונלית בין group\_id לשאר התכונות הכתובות בת"פ 2.**

**group\_id הוא אינו מפתח ראשי של הטבלה Book, לכן יש לנרמל את טבלה זו ע"י הפירוק הבא:**

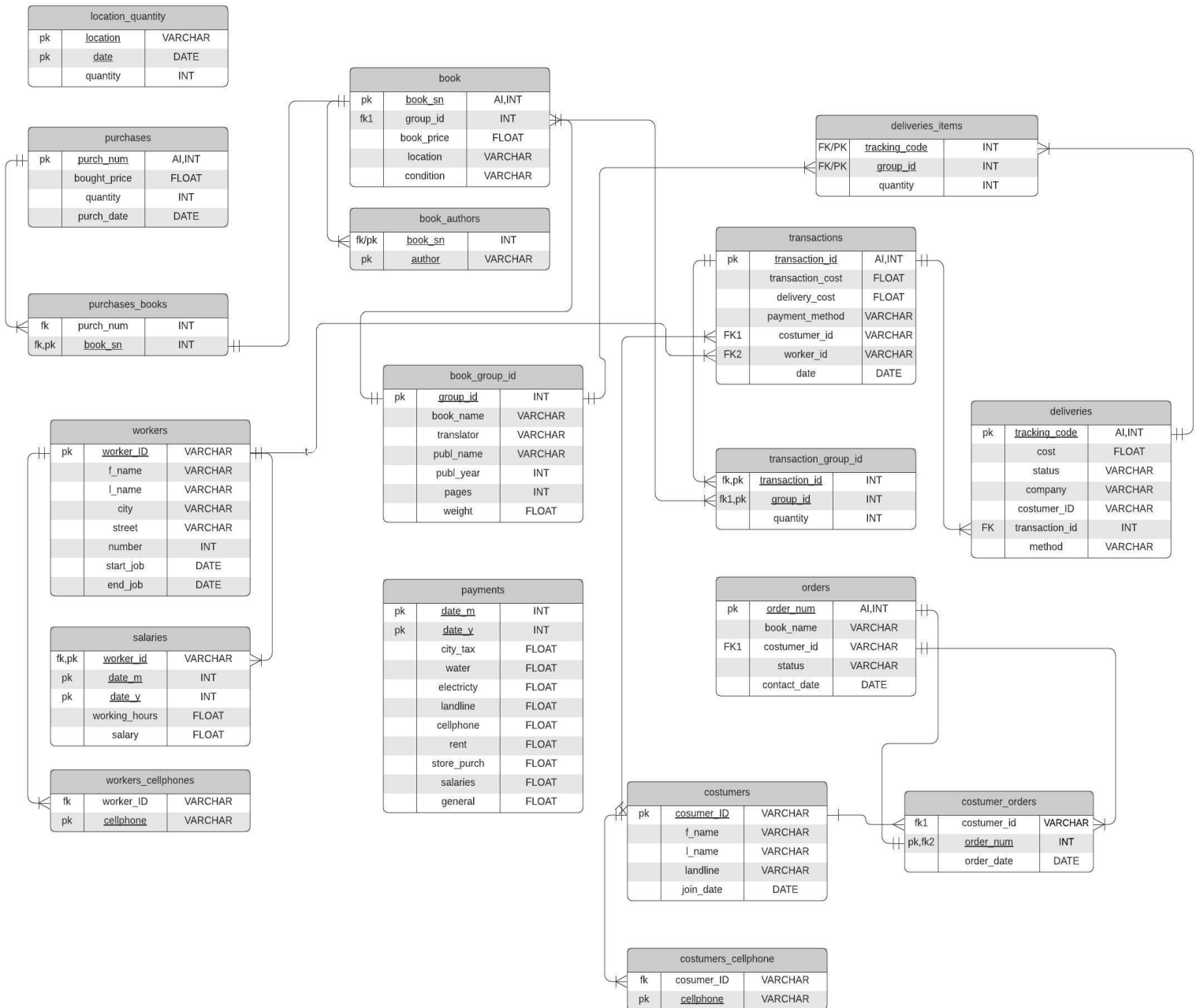
**Book-**

<u>Book_sn</u>	condition	location	book_price	group_id
----------------	-----------	----------	------------	----------

**Book\_group\_id**

<u>Book_name</u>	translator	Publ_name	Publ_year	pages	weighth	<u>group_id</u>
------------------	------------	-----------	-----------	-------	---------	-----------------

## 5. סכמה של בסיס הנתונים לאחר נרמול ל 3NF:



## 6. תוכן כל הטבלאות:

Book

book_sn	group_id	book_price	location	condition
1	3	25	storage	bn
2	4	36	store	n
3	5	245	store	b
4	6	100	storage	o
11	6	120	store	bn
12	7	80	store	m
13	8	25	storage	n
14	12	50	storage	bn
15	13	60	store	n
16	14	100	store	b
20	15	50	store	bn
21	3	50	store	n
22	16	50	store	n
23	17	60	store	bn
24	18	25	storage	n
25	19	40	store	n

Book\_author

book_sn	author
1	guy
1	opal
2	dekel
3	sivan
4	danit
11	keren
12	itamar
13	ela
14	gali
15	yoav
16	efrat

Costumer\_orders

order_num	costumer_id	order_date
1	234535678	2020-03-11
2	325754635	2020-07-11
3	346478432	2020-11-11
4	346547756	2019-03-11
5	346673167	2019-03-15
6	457457547	2019-03-09
7	485929344	2019-08-11
8	547856743	2019-12-11
9	654715246	2019-10-01
10	774564574	2019-03-28

Costumer\_cellphones

costumer_id	cellphone
234535678	0527485927
325754635	0563857245
346478432	0512485242
346547756	0523758292
346673167	0523412468
457457547	0534522214
485929344	0523422411
547856743	0521144459
654715246	0525252367
774564574	0522559099

Book\_group\_id

group_id	book_name	translator	publisher_name	publisher_year	pages	weight
3	harry	poter	ron	2012	345	0.56
4	good sql	guy	opal	2013	435	1.3
5	coding	dekel menashe	kineret	2017	14	0.1
6	water flow	ron kivson	orgad	1987	368	0.6
7	rise of machines	arnold	robo cop	1998	647	1
8	hot in the sun	sunny bunny	Mr rabbit	1234	612	1.2
9	algebra	rick	morty	1683	251	0.29
12	coding	korman	guy	1984	231	0.89
13	big burgers	sponge bob	patrick	2020	35	0.34
14	cactus	roses	violets	1945	456	2.8
15	harry	pit	ema	2012	345	0.56
16	po the bear	bell	adir	2013	200	0.9
17	tarazn	guy	opal	2014	210	0.23
18	cinderella	peretz	moshe	2015	220	0.2
19	mr bean	bibi	sara	2015	300	0.12



costumers

costumer_id	f_name	l_name	landline	join_date
234535678	gil	isban	NULL	2019-12-05
325754635	gary	anan	NULL	2019-03-05
346478432	sami	satlan	032464910	2019-10-05
346547756	hanni	matan	094536593	2020-09-18
346673167	tami	mutshan	NULL	2020-04-06
457457547	gili	eilan	NULL	2020-10-26
485929344	gal	hanan	NULL	2020-10-16
547856743	ami	eran	NULL	2018-04-07
654715246	gon	shanan	NULL	2018-11-07
774564574	gitit	nitsan	NULL	2018-10-07

purchases

purch_num	bought_price	quantity	purch_date
1	13.5	1	2019-01-20
2	45	1	2019-01-01
3	25	1	2020-06-01
4	25	1	2019-10-06
5	30	1	2016-10-01
6	35	1	2019-01-23
7	40	1	2019-01-06
8	45	1	2019-01-01
9	50	1	2017-01-01
10	55	1	2018-01-01

deliveries

tracking_code	cost	del_status	company	costumer_id	transaction_id	method
1	15	deliverd	post	234535678	1	mail
2	15	deliverd	post	325754635	2	r_delivery
3	15	deliverd	xpress	346478432	3	delivery
4	60	deliverd	xpress	346547756	4	f_delivery
5	15	sent	xpress	346673167	5	pickup
6	15	deliverd	post	457457547	7	f_delivery
7	20	deliverd	post	457457547	7	delivery
8	30	deliverd	post	547856743	8	regular
9	15	deliverd	post	547856743	8	mail
10	15	deliverd	xpress	774564574	10	mail
18	77	deliverd	xpress	234535678	12	mail
20	15	deliverd	xpress	547856743	99	mail

deliveries\_items

tracking_code	group_id	quantity
1	3	1
2	4	1
3	8	1
4	8	1
5	5	1
6	12	2
7	3	1
8	5	1
9	3	1
9	4	1
9	7	1
10	5	1
18	3	1
18	15	1

location\_quantity

location	date	quantity
storage	2016-10-01	0
storage	2019-01-01	3
storage	2019-02-01	3
storage	2019-04-01	4
store	2016-10-01	1
store	2017-01-01	2
store	2018-01-01	1
store	2019-01-01	5
store	2019-02-01	5
store	2020-12-01	6

orders

order_num	book_name	costumer_id	o_status	contact_date
1	aladin	234535678	waiting	NULL
2	ami tami	325754635	waiting	NULL
3	home alone	346478432	waiting	NULL
4	data analisis	346547756	waiting	NULL
5	tom and jerry	346673167	waiting	NULL
6	mr bean	457457547	updated	2019-04-19
7	cinderella	485929344	updated	2019-04-19
8	tarazn	547856743	updated	2019-10-19
9	po the bear	654715246	updated	2019-04-13
10	palm beach	774564574	waiting	NULL

## payments

salaries	rent	store_purch	cellphone	landline	electricity	water	city_tax	date_y	date_m	general
0	1500	50	30	20	240	90	600	2017	1	NULL
0	1500	55	30	20	240	90	600	2018	1	NULL
0	1500	0	30	20	240	90	600	2018	2	NULL
0	1500	178.5	30	20	240	90	600	2019	1	NULL
0	1500	0	30	20	240	90	600	2019	2	NULL
0	1500	25	30	20	240	90	600	2019	10	NULL
2250	1500	13.5	30	20	250	100	600	2020	3	NULL
2000	1500	0	30	20	300	120	600	2020	4	NULL
1000	1500	0	30	20	240	90	600	2020	5	NULL
0	1500	25	30	20	240	90	600	2020	6	NULL

## Purchases\_books

book_sn	purch_num
1	1
2	2
3	3
4	4
11	5
12	6
13	7
14	8
15	9
16	10

## salaries

worker_id	date_m	date_y	working_hours	salary
125436341	3	2020	5	250
125436341	4	2020	10	500
125436341	5	2020	5	250
353253646	3	2020	10	500
353253646	4	2020	10	500
353253646	5	2020	5	250
355647686	3	2020	20	1000
355647686	4	2020	10	500
355647686	5	2020	5	250
378829566	3	2020	10	500
378829566	4	2020	10	500
378829566	5	2020	5	250

## Transaction\_group\_id

group_id	transaction_id	quantity
3	1	1
3	7	1
3	12	1
4	2	1
5	8	1
7	8	1
7	9	1
8	3	1
8	4	1
9	6	1
12	7	2
14	10	1
14	99	1
15	12	1
16	100	1
17	101	1

## Transactions

transaction_id	transaction_cost	delivery_cost	payment_method	costumer_id	worker_id	t_date
1	40	15	credit	234535678	125436341	2020-07-13
2	35	15	credit	325754635	125436341	2019-01-13
3	80	15	credit	346478432	125436341	2019-10-13
4	250	60	credit	346547756	355647686	2019-10-19
5	33.5	15	bit	346673167	355647686	2017-12-03
6	29	35	bit	457457547	355647686	2020-04-19
7	150	0	transfer	485929344	378829566	2019-05-16
8	100	45	transfer	547856743	378829566	2019-05-11
9	55	0	credit	774564574	353253646	2019-03-28
10	50	15	bit	774564574	353253646	2019-09-06
12	60	77	bit	234535678	353253646	2019-09-28
99	22	12	credit	547856743	353253646	2019-09-16
100	60	0	credit	547856743	353253646	2019-10-27
101	50	0	credit	654715246	353253646	2019-12-29

Worker\_cellphones

worker_id	cellphone
125436341	0526724859
214214214	0532432552
235235526	0572354223
236876315	0532532523
353253646	0535235235
355647686	0515905034
378829566	0591312436
436575832	0503526345
794572352	0579235457
956365745	0511346774

Workers

worker_id	f_name	l_name	city	street	house_num	start_date	end_date
125436341	ran	boker	holon	gardom	6	2020-01-01	NULL
214214214	mark	shagal	tel aviv	golomb	78	2018-01-01	2018-05-01
235235526	michal	ronen	herzeliya	fabrigat	54	2018-04-01	2018-12-01
236876315	rachel	cohen	tel aviv	grfen	12	2017-10-01	2018-01-01
353253646	yaron	asher	savyon	bedner	34	2019-12-23	NULL
355647686	eyal	golan	givatyim	ben goryon	13	2018-06-12	NULL
378829566	cohen	raz	rehovot	zalman	3	2020-04-18	NULL
436575832	bibi	netanyaho	ramat gan	bialik	84	2019-03-01	2020-02-02
794572352	yarden	biran	bat yam	noah	2	2016-05-05	2017-05-05
956365745	shuki	zikri	yeroshalim	zabotinski	5	2018-09-09	2018-10-09

**ב.6 עבור כל פעולה המצוינת בסעיף 3.6 במסמך העבודה קוד SQL המממש אותה והפעולה באלגברת יחסים:**

**q1:**

```
SELECT location
from book
where group_id in (select group_id from book_group_id where book_name = '?????');
```

**אלגברת יחסים q1**

$\pi_{location} (\sigma_{book\_name = '????'} (book \bowtie_{book\_group\_id.group\_id = book.group\_id} (book\_group\_id)))$

**q2:**

```
select costumer_id , f_name, l_name , join_date
from costumers
where join_date = ( select min(join_date) from costumers);
```

**אלגברת יחסים q2**

$\pi_{costumer\_id, f\_name, l\_name, join\_date} (\sigma_{join\_date = \sigma_{MIN(join\_date)}(costumers)})$

**q3:**

```
select book_name
from(
select group_id
from(
select book_sn
from(
select purch_num
from purchases
where purch_date = ( select min(purch_date) from purchases) )as sub
```

```

join purchases_books
on sub.purch_num = purchases_books.purch_num) as secondsub
join book
on book.book_sn = secondsub.book_sn) as thirdsub
join book_group_id
on book_group_id.group_id = thirdsub.group_id

```

### **q3 אלגברת יחסים**

```

 $\pi$  book_name  $\sigma$ (thirdsub  $\leftarrow \rho$ ( $\pi$  group_id
secondsub  $\leftarrow \rho$ (  $\pi$  book_sn
sub  $\leftarrow \rho$ (  $\pi$  purch_num ( $\sigma$ 
purch_date = select min(purch_date) (purchases)))
 $\bowtie$  sub.purch_num = purchases_books.purch_num (purchases_books))
 $\bowtie$  book.book_sn = secondsub.book_sn (book))  $\bowtie$  book_group_id.group_id =
thirdsub.group_id(book_group_id))

```

### **q4:**

```

select o.order_num, o.costumer_id, book_name, o_status, order_date
from orders o
inner join costumer_orders co
on o.order_num = co.order_num
order by contact_date desc;

```

### **q4 אלגברת יחסים**

```

 $\tau$  contact_date desc ( $\pi$  orders.order_num, orders.costumer_id, book_name, o_status, order_date
(orders  $\bowtie$  orders.order_num = costumer_orders.order_num (costumer_orders)))

```

**q5:**

```

select  *,count(*)
from    book_group_id
        join transaction_group_id
            on book_group_id.group_id = transaction_group_id.group_id
        where book_name = '????' ;

```

**אלגברת יחסים q5**

$\sigma (\sigma \text{ book\_name} = \text{'????'} \mid \text{transaction\_group\_id} \bowtie \text{book\_group\_id.group\_id} = \text{transaction\_group\_id.group\_id} \mid \text{book\_group\_id})$

**q6:**

```

select author
from book_authors
join(
select book_sn
from book
join(
select group_id
from transactions
join transaction_group_id
on transactions.transaction_id = transaction_group_id.transaction_id
    where t_date between '2020-04-19' and '2020-12-19') as sub
on book.group_id = sub.group_id )as secondsub
on secondsub.book_sn = book_authors.book_sn
limit 1;

```

אלגברת יחסים q6

```

π author σ(  secondsub←ρ( π book_sn
                sub←ρ( π group_id (σ (transaction_group_id
                transactions.transaction_id = transaction_group_id.transaction_id
                AND t_date between? AND ? (transactions)))
                ⋈ book.group_id = sub.group_id (book))
                ⋈ secondsub.book_sn = book_authors.book_sn (book_authors))

```

q7:

```
select f_name, l_name
from ( select customer_id, count(customer_id) as freqvalue
      from transactions
      group by customer_id
      order by freqvalue desc
      limit 3) as subquery
      join customers
      on subquery.customer_id = customers.customer_id;
```

אלגברת יחסים q7

```

π f_name,l_name( subquery←p( τ freqvalu( γ costumer_id( rownum()>0 ∧ rownum()<3(σ costumer_id,
freqvalue←p(count(costumer_id))))))
⋈ subquery.costumer_id = costumers.costumer_id (costumers))

```

q8:

```
select *, count(book_name)
from(
    select translator, book_name, group_id
    from book_group_id
```

```

) as secondsubquery
group by book_name
order by count(book_name) desc
limit 1;

```

#### **q8 אלגברת יחסים**

```

τ count(book_name) desc (
  γ book_name (
    rownum()>0 ∧ rownum()<=1 (
      subquery←ρ(π translator,book_name,group_id,
count(book_name)(book_group_id))))

```

#### **q9:**

```

select f_name, l_name, book_name, t_date, transaction_cost
from(
  select transaction_cost, f_name, l_name, t_date, group_id
  from(
    select transaction_cost, f_name, l_name, t_date, transaction_id
    from(
      select costumer_id, f_name, l_name
      from costumers
      where costumer_id = '???????) as subquery
    join transactions
    on subquery.costumer_id = transactions.costumer_id) as secondsub
  join transaction_group_id
  on secondsub.transaction_id = transaction_group_id.transaction_id) as thirdsub
  join book_group_id
  on thirdsub.group_id = book_group_id .group_id
order by t_date desc ;

```



### אלגברת יחסים q9

```
τ t_date desc(thirdsub←ρ(π f_name,l_name,book_name,t_date,transaction_cost
    secondsub←ρ( π transaction_cost, f_name, l_name, t_date, group_id
        firstsub←ρ( π costumer_id, f_name, l_name (σ costumer_id = '?????' ) )
            ⋈ subquery.costumer_id = transactions.costumer_id(transactions))
        ⋈ secondsub.transaction_id = transaction_group_id.transaction_id (transaction_group_id))
    ⋈ thirdsub.group_id = book_group_id .group_id (book_group_id))
```

### q10:

```
select f_name, l_name, costumer_orders.costumer_id, book_name, o_status,
costumer_orders.order_num, contact_date, order_date
    from(
        select f_name, l_name, orders.costumer_id, book_name, o_status, order_num,
        contact_date
        from(
            select f_name, l_name, costumer_id
            from costumers
            where costumer_id = '?????') as subquery
        join orders
            on subquery.costumer_id = orders.costumer_id ) as secondsub
        join costumer_orders
            on secondsub.order_num = costumer_orders.order_num
        order by order_date desc;
```

### אלגברת יחסים q10

```
τ order_date desc (
    π f_name, l_name, costumer_orders.costumer_id, book_name, o_status,
    costumer_orders.order_num, contact_date, order_date (
        secondsub←ρ(π f_name, l_name, orders.costumer_id, book_name, o_status,
        order_num, contact_date
            subquery←ρ(π f_name, l_name, costumer_id (σ costumer_id = '?????'))
```

```

    ⋈ subquery.costumer_id = orders.costumer_id(orders))
    ⋈ secondsub.order_num = costumer_orders.order_num (costumer_orders)))

```

**q11:**

```

select sum(weight) as booksTotalWeight
from(
    select weight, book_name
    from book_group_id
    where book_name = '?????' or book_name='????'    //(more books will be added if wanted)
    group by book_name) as totalweight ;

```

**אלגברת יחסים q11**

```

π bookstotalweight←ρ(sum(weight))(
    totalweight←ρ(
        γ book_name(
            π weight,book_name(
                σ book_name = '?????' ∨ book_name = '?????' ∨ ... (book_group_id))))))

```

**q12:**

```

select tracking_code, cost, del_status, company, costumer_id, method, deliveries.transaction_id
from(
    select transaction_id, count(*)
    from(
        select tracking_code, transaction_id
        from deliveries
        where transaction_id in(
            select transaction_id
            from(

```

```

select costumer_id, f_name, l_name
from costumers
where costumer_id = '???????' as subquery
join transactions
on subquery.costumer_id = transactions.costumer_id ) as secondsub
group by transaction_id
having count(*) >1 ) as thirdsub
join deliveries
on thirdsub.transaction_id = deliveries.transaction_id ;

```

### **q12** אלגברת יחסים

```

π author σ( secondsub ← ρ( π book_sn
sub ← ρ( π group_id (σ (transaction_group_id)
transactions.transaction_id = transaction_group_id.transaction_id
AND t_date between ? AND ? (transactions)))
⋈ book.group_id = sub.group_id (book))
⋈ secondsub.book_sn = book_authors.book_sn (book_authors))

```

### **q13:**

```

select del_status from deliveries where tracking_code = '???????' ;

```

### **q13** אלגברת יחסים

```

π del_status (σ tracking_code = '5' (deliveries))

```

### **q14:**

```

select company, sum(cost)
from(
select tracking_code, company, cost

```

```

from(
    select transaction_id
    from transactions
    where t_date between '????-??-01' and '????-??-30') as subquery
join deliveries
on subquery.transaction_id = deliveries.transaction_id ) as secondsub
where company = 'xpress' ;

```

#### **q14 אלגברת יחסים**

```

π company, sum(cost)(
    σ company = 'xpress'(
        secondsub ← ρ(
            π tracking_code, company, cost
            subquery ← ρ (
                π transaction_id(
                    σ t_date between '2019-09-01' and '2019-09-30'(transactions)))
            ⋈ subquery.transaction_id = deliveries.transaction_id(deliveries))))

```

#### **q15:**

```

select payment_method, sum(transaction_cost)
from(
    select transaction_cost, payment_method, t_date
    from transactions
    where month(t_date) = '??' and year(t_date) = '????') as subquery
where payment_method = 'bit' ;

```

### אלגברת יחסים q15

```
π payment_method,sum(transaction_cost)(
    σ payment_method = 'bit'(
        subquery←ρ(
            π transaction_cost, payment_method, t_date(
                σ month(t_date) = '????' and year(t_date) = '????' (transactions))))))
```

### q16:

```
select transaction_id, t_date, transaction_cost
from(
    select transaction_id, t_date, transaction_cost
    from transactions
    where t_date < current_date()
    and t_date > current_date() - interval 1 year) as secondsub
where transaction_cost > (select sum(transaction_cost)/12
from(
    select t_date, transaction_cost
    from transactions
    where t_date < current_date()
    and t_date > current_date() - interval 1 year ) as subquery);
```

### אלגברת יחסים q16

```
π transaction_id, t_date, transaction_cost(
    σ where transaction_cost > subquery←ρ (
        π sum(transaction_cost)/12(
            π t_date, transaction_cost (
                σ t_date < current_date() and t_date > current_date() - interval 1 year
                (transactions))))))
```

```

secondsub ← ρ(
    π transaction_id, t_date, transaction_cost (
        σ t_date < current_date() and t_date > current_date() - interval 1 year
        (transactions))))

```

**q17:**

```

select company, count(*)
from(
    select transaction_id
    from transactions
    where t_date < current_date() and t_date > current_date() - interval 1 year )as subquery
join deliveries
    on subquery.transaction_id = deliveries.transaction_id
group by company ;

```

**אלגברת יחסים q17**

```

γ company (
    π company, count(*)subquery ← ρ(
        π transaction_id(
            σ t_date < current_date() and t_date > current_date() - interval 1 year
            (transactions)))
    ⋈ subquery.transaction_id = deliveries.transaction_id (deliveries))

```

**q18:**

```
select deliveries.tracking_code, group_id, book_name, cost,
del_status, company, costumer_id, transaction_id, method
from(
    select book_group_id.group_id, tracking_code, book_name
    from (
        select group_id, deliveries_items.tracking_code
        from (
            select tracking_code, COUNT(tracking_code) AS NumOccurrences
            from deliveries_items
            group by tracking_code
            having ( COUNT(tracking_code) > 1 )) as subquery
        join deliveries_items
        on subquery.tracking_code = deliveries_items.tracking_code) as secondsub
    join book_group_id
    where secondsub.group_id = book_group_id.group_id) as thirdsub
join deliveries
on thirdsub.tracking_code = deliveries.tracking_code ;
```

**אלגברת יחסים q18**

$\pi$  deliveries.tracking\_code, group\_id, book\_name, del\_status, company, costumer\_id, transaction\_id, method

thirdsub  $\leftarrow$   $\rho$ (

$\sigma$  secondsub.group\_id = book\_group\_id.group\_id(

$\pi$  book\_group\_id.group\_id, tracking\_code, book\_name

secondsub  $\leftarrow$   $\rho$ (

$\pi$  group\_id, deliveries\_items.tracking\_code

subquery  $\leftarrow$   $\rho$ (

```

        γ tracking_code(
            π
            NumOccurrences ← ρ(COUNT(tracking_code)), tracking_code(
                σ COUNT(tracking_code) > 1
                (deliveries_items)))
        ⋈ subquery.tracking_code =
        deliveries_items.tracking_code(deliveries_items))
        ⋈ (book_group_id)))
        ⋈ thirdsub.tracking_code = deliveries.tracking_code (deliveries)

```

**q19:**

```

select c.costumer_id, f_name, l_name, landline, join_date, cellphone
    from costumers c
    join costumer_cellphones cc
        on cc.costumer_id = c.costumer_id
    join transactions t
        on t.costumer_id = c.costumer_id
    where t_date < NOW() - INTERVAL 2 YEAR;

```

**אלגברת יחסים q19**

```

π c.costumer_id, f_name, l_name, landline, join_date, cellphone(
    σ t_date < NOW() - INTERVAL 2 YEAR ( costumers c
    ⋈ cc.costumer_id = c.costumer_id(costumer_cellphones cc)
    ⋈ t.costumer_id = c.costumer_id (transactions t)))

```



**q20:**

```

select fourthsub.costumer_id, f_name, l_name, cellphone
from(
select costumer_id
from(
select costumer_id, group_id
from(
select contact_date, costumer_id, book_name
from(
select costumer_id, book_name, contact_date
from orders
where o_status = 'updated' and contact_date is not null) as subquery
where year(NOW()) > year(subquery.contact_date)
or month(NOW()) > month(subquery.contact_date)
or month(NOW()) = month(subquery.contact_date) and day(NOW()) >
day(subquery.contact_date)) as secondsub
join book_group_id
where secondsub.book_name = book_group_id.book_name) as thirdsub
join transaction_group_id
where thirdsub.group_id = transaction_group_id.group_id) as fourthsub
join( select cost.costumer_id, cellphone, f_name, l_name
from costumers as cost
join costumer_cellphones as cell
on cost.costumer_id = cell.costumer_id ) as thejoin
on fourthsub.costumer_id = thejoin.costumer_id ;

```

**אלגברת יחסים q20**

```

π fourthsub.costumer_id, f_name, l_name, cellphone
forthsub ← ρ(

```

```

σ thirdsub.group_id = transaction_group_id.group_id(
    π costumer_id
    thirdsub ← ρ(
        σ secondsub.book_name = book_group_id.book_name(
            π costumer_id, group_id
            secondsub ← ρ(
                σ year(NOW()) > year(subquery.contact_date)
                    or month(NOW()) > month(subquery.contact_date)
                    or month(NOW()) = month(subquery.contact_date) and
day(NOW()) > day(subquery.contact_date)(
                    π contact_date, costumer_id, book_name (
                        subquery ← ρ(
                            σ o_status = 'updated' and
contact_date is not null (orders))))))
                ⋈ book_group_id))
            ⋈ transaction_group_id))
        ⋈ fourthsub.costumer_id = thejoin.costumer_id thejoin ← ρ(
            π cost.costumer_id, cellphone, f_name, l_name(
                ⋈ cost.costumer_id = cell.costumer_id cost ← ρ(costumers)))

```

#### **q21:**

```

select *
from locations_quantity
where location = 'storage';

```

#### **אלגברת יחסים - q21**

```

σ(cation = 'storage'(locations_quantity))

```

**q22-a:**

```

select count(book_sn) as amountOfBooks , sum(bought_price) as totalCost
from purchases p
      join purchases_books pb
            on pb.purch_num = p.purch_num
      where purch_date between '????-??-??' and '????-??-??';

```

**q22-a - אלגברת יחסים**

$\Pi$  sum(bought\_price), count(book\_sn)  $\sigma((\text{purchases}) \bowtie \text{pb.purch\_num} = \text{p.purch\_num} (\text{purchases\_books}) \text{ AND purch\_date between ? AND ?})$

**q22-b:**

```

select *
from(
      select sum(bought_price)
      from purchases
      where month(purch_date) = '??' and year(purch_date) = '????') as subquery
      join( select sum(transaction_cost)
            from transactions
            where month(t_date) = '??' and year(t_date) = '????') as secondsub ;

```

**q22-b - אלגברת יחסים**

$\sigma(\Pi(\text{sum}(\text{bought\_price})$   
 subquery  $\leftarrow P(\sigma(\text{purchases.month}(\text{purch\_date}) = ? \text{ and } \text{purchases.year}(\text{purch\_date}) = ?)) \bowtie$   
 secondsub  $\leftarrow (P \Pi \text{sum}(\text{transaction\_cost}) \text{ month}(\text{t\_date}) = ? \text{ AND } \text{year}(\text{t\_date}) = ? (\text{ transactions}))))$

**q23:**

select

(select avg(transaction\_cost)

from transactions

where month(t\_date) ='01'

and year(t\_date) ='2019') as January,

(select avg(transaction\_cost)

from transactions

where month(t\_date) ='02'

and year(t\_date) ='2019') as February,

(select avg(transaction\_cost)

from transactions

where month(t\_date) ='03'

and year(t\_date) ='2019') as March,

(select avg(transaction\_cost)

from transactions

where month(t\_date) ='04'

and year(t\_date) ='2019') as April,

(select avg(transaction\_cost)

from transactions

where month(t\_date) ='05'

and year(t\_date) ='2019') as May,

(select avg(transaction\_cost)

from transactions

where month(t\_date) ='06'

and year(t\_date) ='2019') as June,

(select avg(transaction\_cost)

```

from transactions
where month(t_date) ='07'
and year(t_date) ='2019') as July,
(select avg(transaction_cost)
from transactions
where month(t_date) ='08'
and year(t_date) ='2019') as August,
(select avg(transaction_cost)
from transactions
where month(t_date) ='09'
and year(t_date) ='2019') as September,
(select avg(transaction_cost)
from transactions
where month(t_date) ='10'
and year(t_date) ='2019') as October,
(select avg(transaction_cost)
from transactions
where month(t_date) ='11'
and year(t_date) ='2019') as November,
(select avg(transaction_cost)
from transactions
where month(t_date) ='12'
and year(t_date) ='2019') as December ;

```

### אלגברת יחסים-23q

$\pi$

```

January ← ρ(π avg(transaction_cost)(σ month(t_date) ='???' and year(t_date) ='???')(transactions)),
Febuary ← ρ(π avg(transaction_cost)(σ month(t_date) ='???' and year(t_date) ='???')(transactions)),
March ← ρ(π avg(transaction_cost)(σ month(t_date) ='???' and year(t_date) ='???')(transactions)),

```

April  $\leftarrow \rho(\pi \text{ avg}(\text{transaction\_cost})(\sigma \text{ month}(\text{t\_date}) = '???' \text{ and year}(\text{t\_date}) = '???')(\text{transactions})),$   
 May  $\leftarrow \rho(\pi \text{ avg}(\text{transaction\_cost})(\sigma \text{ month}(\text{t\_date}) = '???' \text{ and year}(\text{t\_date}) = '???')(\text{transactions})),$   
 June  $\leftarrow \rho(\pi \text{ avg}(\text{transaction\_cost})(\sigma \text{ month}(\text{t\_date}) = '???' \text{ and year}(\text{t\_date}) = '???')(\text{transactions})),$   
 July  $\leftarrow \rho(\pi \text{ avg}(\text{transaction\_cost})(\sigma \text{ month}(\text{t\_date}) = '???' \text{ and year}(\text{t\_date}) = '???')(\text{transactions})),$   
 August  $\leftarrow \rho(\pi \text{ avg}(\text{transaction\_cost})(\sigma \text{ month}(\text{t\_date}) = '???' \text{ and year}(\text{t\_date}) = '???')(\text{transactions})),$   
 September  $\leftarrow \rho(\pi \text{ avg}(\text{transaction\_cost})(\sigma \text{ month}(\text{t\_date}) = '???' \text{ and year}(\text{t\_date}) = '???')(\text{transactions})),$   
 October  $\leftarrow \rho(\pi \text{ avg}(\text{transaction\_cost})(\sigma \text{ month}(\text{t\_date}) = '???' \text{ and year}(\text{t\_date}) = '???')(\text{transactions})),$   
 November  $\leftarrow \rho(\pi \text{ avg}(\text{transaction\_cost})(\sigma \text{ month}(\text{t\_date}) = '???' \text{ and year}(\text{t\_date}) = '???')(\text{transactions})),$   
 December  $\leftarrow \rho(\pi \text{ avg}(\text{transaction\_cost})(\sigma \text{ month}(\text{t\_date}) = '???' \text{ and year}(\text{t\_date}) = '???')(\text{transactions}))$

#### **q24:**

```

select f_name, l_name , salary
from workers a
      join salaries w
            on w.worker_id = a.worker_id
where w.worker_id = '???????' and date_m = '??' and date_y = '?????';

```

#### **q24- אלגברת יחסים**

$\Pi f\_name, l\_name, salary \sigma((\text{workers}) \bowtie \text{workers.worker\_id} = \text{salaries.worker\_id}(\text{salaries}) \text{ AND } \text{workers.worker\_id} = ? \text{ And } \text{date\_m} = ? \text{ And } \text{date\_y} = ?)$

#### **q25:**

```

select w.f_name, l_name, count(t.worker_id) as transactionsamount
from transactions t
      join workers w
            on w.worker_id = t.worker_id
where month(t_date) = '??' and year(t_date) = '????'
group by t.worker_id

```

order by transactionsamount desc

limit 1;

#### **q25 - יחסים**

A  $\leftarrow \Pi w.f\_name, l\_name, count(t.worker\_id) \sigma((transactions) \bowtie workers.worker\_id = transactions.worker\_id (workers) \text{ AND } month(t\_date) = ? \text{ And } year(t\_date) = ?)$

B  $\leftarrow (worker\_id) \gamma(A)$

C  $\leftarrow \tau(transactionsamount \text{ desc})(B)$

## רשימת השאלות הפתוחות שעלו בזמן תכנון ומימוש האפליקציה:

1. איך מבדילים בין ספר שקיים במלאי בהווה לבין ספר שהיה קיים במלאי בעבר?  
תשובה – לכל ספר מספר סידורי ייחודי. בנוסף לכל ספר משויך group id שמבטא את השתייכות הספר למהדורה ספציפית.  
לכן, בבדיקה האם ספר נמצא במלאי ייתכנו שתי אפשרויות:  
א. יש לספר מופע בקבוצת Book, כלומר קיים ספר בעל מספר סידורי X השייך לקבוצה זו ונמצא במלאי בהווה.  
ב. אין לספר מופע בקבוצת Book, כלומר אין מספר סידורי עבור ספר זה, אך המהדורה אליה הספר משתייך קיימת – (היה ספר כזה בעבר אך בהווה אינו נמצא במלאי).  
2. איך מזינים ספר במערכת?  
תשובה – ישנן שתי אפשרויות:  
א. מהדורת הספר קיימת במסד הנתונים – כלומר יש להוסיף מופע של הספר בטבלת Book בלבד ולשייך אותו למהדורה הרלוונטית על ידי שדה group\_id.  
ב. מהדורת הספר לא קיימת במסד הנתונים – יש להוסיף מופע של הספר ב Book ואת מהדורת הספר בטבלת group\_id ולקשר בין הטבלאות על ידי עדכון שדה group\_id בטבלת Book.  
3. איך מוחקים ספר במערכת?  
תשובה – על מנת לשמור על תאימות לסעיף 1, מחיקת ספר תתבצע אך ורק על ידי מחיקת מופע הספר בטבלת Book.  
4. שאילתה 6 – "מי הסופר הכי נקרא בתווך התאריכים X עד Y?"  
שאלה – מה מגדיר סופר הכי נקרא?  
תשובה – הסופר שספריו נמכרו הכי הרבה בתווך התאריכים הנתונים.  
5. שאילתה 8 – "מי הוא הספר עם מספר התרגומים הגדול ביותר שקיים כרגע במלאי?"  
שאלה – האם הכוונה במספר תרגומים גדול ביותר הינה באופן כללי בעולם או מתוך מספר התרגומים שקיימים בחנות / היו בעבר ברשות החנות, עבור הספר המצוין.  
תשובה – הספר עם מספר התרגומים הגדול ביותר הינו הספר עם מספר התרגומים המקסימלי מתוך מספר התרגומים שקיימים בחנות / היו בעבר ברשות החנות, עבור הספר המצוין.  
6. שאילתה 23 – "ממוצע עסקאות שנתי בחתך חודשי"  
שאלה – האם הכוונה בממוצע זה לממוצע עסקאות עבור כל חודש בשנה X.  
או ממוצע עסקאות בכל חודש עבור כל השנים בהן החנות פעילה.  
תשובה – הכוונה בממוצע זה לממוצע עסקאות עבור כל חודש בשנה X.