עבודת פרויקט SQL

מגישה: דנה דניאלה אלוני 207907742

שנה ב' – סמסטר ב'

מרצה: מר יצחק נודלר

סכמה ראשונית –

טבלת לקוח- Costumer

Id_costumer	First_name	Last_name	Id_card	Mobile_number	Telephone_number	Join_date

טבלת משלוח- Delivery

Id_delivery	Company	Delivery_type	Price_per_kg

טבלת עובד - Employee

Id_employees	Full_name	Mobile_number	address	Join_date	Currently_working	Leave_date

טבלת משכורות עובדים- Employees salary

Salary_date	Hourly_price	Total_hours	Salary_sum	Id_employees

טבלת תשלומי חנות- Store payment

Payment_date	water	electricity	Property_tax	Mobile_phones	Telephones	Rent	Other

טבלת ספר - Book

Id_ books	title	Author_ name	Translator_ name	Publisher_ name	Year_of_ distribution	Pages_ count	condition	Location	Arrival_ date	Price	Wight

טבלת הזמנות – Orders

Id_orders	date	status	Id_costumer	Transaction_number	Id_books	Last_contact

טבלת רכישת חנות- Store purchase

Id_store_purchases	Date	Id_books	Price	Origin

טבלת עסקה – Transaction

id_ transaction	Transaction _number	Date	Id_ costumers	Id_ orders	Id_books	Total_ wight	Books_ price	Delivery_ price	Total_ price	Id_ delivery	Payment_ type	Tracking_ number	Shipment _ address	Id_ employees	Payment _status	Delivery _status

רשימת תלויות פונקציונליות:

1. בטבלת Books יש לנו תלות פונקציונלית בין ה-id book לבין ה-title , לכל ספר יש id משלו.

2.בטבלת Customers יש לנו תלות פונקציונלית בין ה-id customer לבין ה-first name וה-last name, לכל לקוח יש id משלו.

3.בטבלת delivery יש לנו תלות פונקציונלית בין ה-id delivery לבין חברת המשלוח(company), לכל סוג משלוח יש id משלו.

4. בטבלת employees יש לנו תלות פונקציונלית בין ה-id employees לבין ה-full name של כל עובד בחברה, לכל עובד יש Id משלו.

5. בטבלת employees salary יש תלות פונקציונלית בין תאריך המשכורת(salary date) לבין ה-id employees, לכל עובד יש משכורת שונה בתאריכים שונים.

6. בטבלת orders יש לנו תלות פונקציונלית בין ה-id orders לבין id customers, transaction number, id books, לכל הזמנה יש מספר משלה שצריכה להיות בתלות עם רשימה זו, לפי כל הנתונים האלו ניתן לבצע הזמנה, לכל הזמנה יהיו פרטים שונים ברשימה.

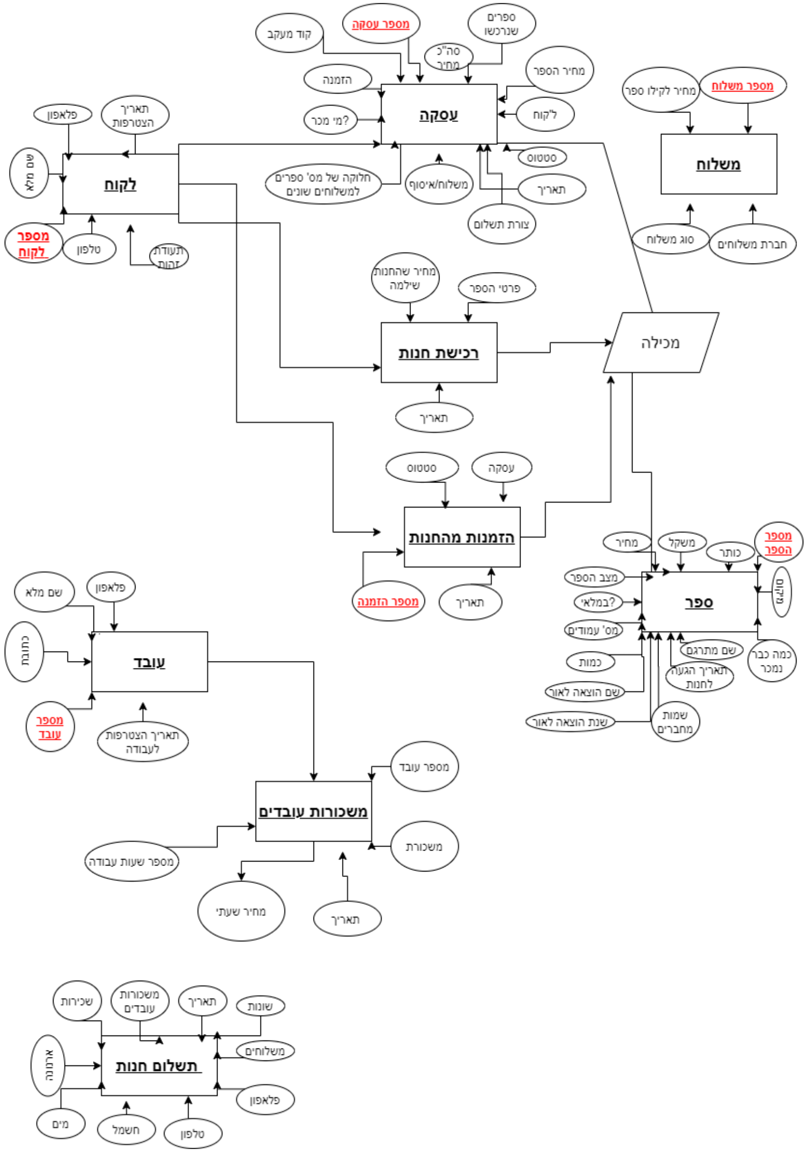
7. בטבלת store purchases יש לנו תלות פונקציונלית בין ה- id store purchases לבין ה-id books, הרי לכל ספר יש מספר id משלו וכל רכישת חנות מבוצעת על ידי ספר אחד או ספרים אחרים.

8. בטבלת transaction יש לנו תלות פונקציונלית בין ה-id transaction לבין ה- id customers, id orders, id books, transaction number, id delivery, id employees, נשים לב שעל מנת לבצע עסקה עלינו למלא את כל הפרטים האלו שבאים לידי ביטוי בטבלאות אחרות, ובהן יש מידע לדוגמה על ההזמנה(על ידי id orders) ומידע על המשלוח (על ידי id delivery), לכל עסקה יש את הפרטים השונים שלה.

נוכל לראות את ישויות הדיאגרמה-
1. עסקה, 2. לקוח, 3. משלוח, 4.רכישת חנות, 5.
הזמנות מהחנות 6.ספר, 7. עובד 8. משכורות עובדים,
9. תשלומי חנות.
לכל ישות ישנן תכונות מוגדרות.

ישנן ישויות שהן חלשות(כלומר ללא מפתח)-
כמו משכורות עובדים ותשלום חנות
הdiscriminator של ישות משכורות עובדים הינו ה- id
.employee
הdisfromination של ישות תשלום חנות הינו ה-
.payment date

נוכל לראות את הקשרים שבין הישויות-
רכישת חנות, הזמנות מהחנות, ועסקה מכילים את
ישות -ספר.

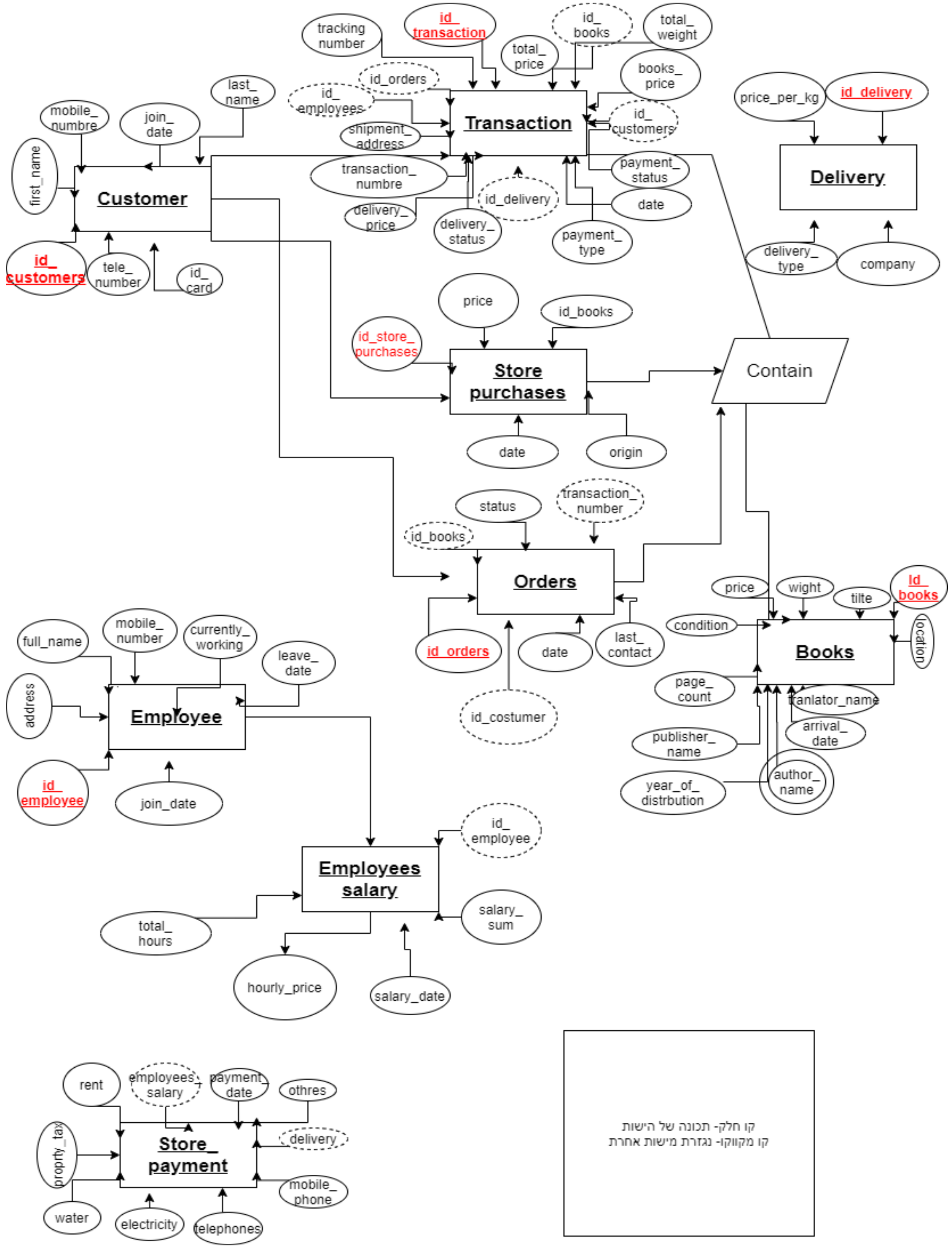


הנחות שהונחו :

1. כל ספר שהחנות רכשה יופיע ברשימת הספרים ומקומו יהיה בחנות או במחסן.
2. במידה ולקוח קנה ספר, מיקום הספר יופיע כ"לא במלאי" – (out of stock)
3. ישנן שתי הרשאות לאפליקציה – מנהל ועובד בחנות. למנהל יש הרשאת מנהל (שכולל בו את כל האפשרויות של מנהל) לכל הטבלאות ואילו לעובד יש הרשאה לכל הטבלאות מלבד טבלת תשלומי חנות, טבלת משכורות עובדים וטבלת עובדים.
4. לספר יש מחבר אחד.
5. ללקוח יש מספר טלפון אחד ומספר אחד סלולרי.
6. עסקה נחשבת עסקה אפילו אם הלקוח לא שילם.
7. בחנות זו – אין הנחות !
8. ברגע שנוצרה הזמנה, תהיה גם עסקה.
9. לכל עסקה יש id transaction שונה, אך כשזו אותה עסקה מפוצלת למשלוחים שונים ישנו אותו transaction number.

סכמה לאחר נרמול

נוכל לראות בכל הקווים המקווקווים שהם רק
שדות מפתח לכן זהו נרמול מסוג 3NF.



קו חלק- תכונה של הישות
קו מקווקו- נגזרת מישות אחרת

Books - table

weight	price	arrival_date	location	condition	pages_count	year_of_distribution	publisher_name	translator_name	author_name	title	id_books
0.32	20	03/01/2018	Warehouse	good	138	1961	M.Mizrahi	NULL	Shraga Dafni	Dani Din Who Sees And Cannot Be Seen	12647
1.2	35	22/05/2018	Store	Poor	318	1997	Phroza	Gili Bar Hillel	J. K Rolling	Harry Potter and the Philosopher's Stone	12648
0.89	45	22/05/2018	out of stock	As new	223	1997	Bloomsbury	Gili Bar Hillel	J. K Rolling	Harry Potter and the Philosopher's Stone	12649
1.7	50	12/12/2018	out of stock	medium	887	2003	Phroza	Gili Bar Hillel	J. K Rolling	Harry Potter and the Order of the Phoenix	12650
0.12	60	17/01/2019	Warehouse	Brand new	30	1977	Workers' Library	NULL	Miriam Ruth	Yael's house	12651
0.86	54	NULL	out of stock	medium	217	2008	Kinneret Zmora Pavilion	Julius Caesar	Jeff Kinney	Diary of a Wimpy Kid	12652
0.6	50	01/03/2019	Warehouse	medium	187	1999	Harper Collins	Ben Zion Herman	Robin Sharma	The Monk Who Sold His Ferrari	12653
0.11	20	16/04/2019	Store	As new	22	2003	The United Kibbutz	NULL	Tirza Atar	The lion that loved strawberry	12654
1.55	90	21/08/2019	Warehouse	good	772	1999	Opus	Idit Shurer	George R.R. Martin	A Song of Fire and Ice Game of Thrones	12655
1.82	100	30/09/2019	out of stock	Brand new	896	2000	Opus	Yael Sela-Shapiro	George R.R. Martin	A song of fire and ice confronting the kings	12656
0.12	45	30/09/2019	out of stock	medium	30	1977	Workers' Library		Miriam Ruth	Yael's house	12657
0.13	20	24/11/2019	out of stock	Poor	32	1974	Workers' Library	NULL	Marina Rotenberg	a tale of five balloons	12658
1.42	70	03/12/2019	out of stock	As new	498	2005	Small and brown	NULL	Stephanie Meyer	Twilight	12659
1.46	66	12/01/2020	out of stock	good	563	2006	Small and brown	NULL	Stefan Lager	new Moon	12660
0.62	30	NULL	out of stock	As new	87	1949	Publisher Ltd.	Cyrus Brooks	Erich Kasztner	Ora the duplicated	12661
0.87	50	NULL	out of stock	As new	182	2012	Opus	Enlightened score	The Grimm Brothers	Grimm's Fairy Tales	12662
0.64	45	24/06/2020	Store	medium	387	1917	McMillan	Nissim Martin	Lewis Carroll	Alice in Wonderland	12663
1.55	90	01/07/2020	Store	good	772	1999	Opus	Ishyi Ribo	George R.R. Martin	A Song of Fire and Ice Game of Thrones	12664
1.2	80	02/08/2019	out of stock	As new	389	1937	George Allen	Moshe Naomi	John Ronald	The Hobbit	12665
1.2	80	03/08/2019	out of stock	As new	389	1950	Anvin	Yael Achmon	John Ronald	The Hobbit	12666

Customers – table

Join_date	Telephone_number	Mobile_number	Id_card	Last_name	First_name	Id_customers
20/01/2018	97677107	503233702	222	Orr	Daniella	154
06/02/2018	87821546	503233875	333	Orr	Youval	155
17/04/2018	NULL	548951254	434	Hazanush	Nirush	156
31/05/2018	NULL	542178563	895	Dafadi	Liran	157
14/09/2018	87863244	548125448	676	Tahar	Mor	158
15/09/2018	95478312	524345458	874	Aloni	Adi	159
01/10/2018	94154574	545863214	173	Cohen	Itzik	160
01/10/2018	93556741	545877952	965	Ron	Ilana	161
26/10/2018	32615879	554789312	992	Gal	Haim	162
14/12/2018	NULL	550458711	823	Hayak	Ben	163
22/03/2019	40694565	545478877	766	Dank	Ran	164
20/04/2019	84514646	522358787	307	Brola	Bar	165

Delivery – table

price_per_kg	delivery_type	company	id_delivery
0	PickUp	0	1
20	Fast Delivery	Israel Post	2
10	Normal Delivery	Israel Post	3
5	Registered Post	Israel Post	4
10	Collecting point	Xpress	5
25	House Delivery	Xpress	6

Employees - table

leave_date	currently_working	join_date	address	mobile_number	full_name	id_employees
NULL	Yes	01/01/2018	Katzenelson 15 Kfar Saba	503233702	Dana Orr	1
NULL	Yes	18/11/2018	Aharonovitch 17 Kfar Saba	508948199	Nicole Kapleyan	2
NULL	Yes	01/01/2019	Katzenelson 14 Kfar Saba	546598760	Yuval Orr	3
30/11/2019	No	21/04/2019	Hateena 8 Neve Yamin	525801250	Shai Ben David	4
27/11/2019	No	20/05/2019	Bar Kochba 46 Herzliya	526490940	Aviv Kristal	5
NULL	Yes	12/07/2019	Tel Hai 93 Ramat Gan	522263617	Nir Hazan	6
02/01/2020	No	12/07/2019	Tel Hai 77 Kfar Saba	526998721	Lior Basil	7
02/01/2020	No	25/09/2019	Ahuza 18 Raanana	539989899	Judah Halevi	8
18/02/2020	No	11/02/2020	Weizmann 52 Raanana	583357141	Mai Edgar	9
NULL	Yes	01/07/2020	King's Road 75 Hod Hasharon	522589317	Einat Aharoni	10

Employees salary- table

id_employees	salary_sum	total_hours	hourly_price	salary_date
1	11200	140	80	01/02/2018
1	11760	147	80	01/03/2018
1	9840	123	80	01/04/2018
1	14080	176	80	01/05/2018
1	12325	145	85	01/06/2018
1	12325	145	85	01/07/2018
1	13855	163	85	01/08/2018
1	10455	123	85	01/09/2018
1	12920	152	85	01/10/2018
1	12665	149	85	01/11/2018
1	12665	149	85	01/12/2018
1	12665	149	85	01/01/2019
1	12665	149	85	01/02/2019
1	12665	149	85	01/03/2019
1	12665	149	85	01/05/2019
1	12665	149	85	01/04/2019
1	12665	149	85	01/06/2019
1	12665	149	85	01/07/2019
1	12665	149	85	01/08/2019
1	12665	149	85	01/09/2019
1	12665	149	85	01/10/2019
1	12665	149	85	01/11/2019
1	12665	149	85	01/12/2019
1	12665	149	85	01/01/2020
1	12665	149	85	01/02/2020
1	12665	149	85	01/03/2020
1	12665	149	85	01/04/2020
1	12665	149	85	01/05/2020
1	12665	149	85	01/06/2020
1	12665	149	85	01/07/2020
2	1800	60	30	01/12/2018
2	2940	98	30	01/01/2019
2	2940	98	30	01/02/2019

2	2940	98	30	01/03/2019
2	2940	98	30	01/04/2019
2	2940	98	30	01/05/2019
2	2940	98	30	01/06/2019
2	2940	98	30	01/07/2019
2	2940	98	30	01/08/2019
2	2940	98	30	01/09/2019
2	2940	98	30	01/10/2019
2	2940	98	30	01/11/2019
2	2940	98	30	01/12/2019
2	2940	98	30	01/01/2020
2	2940	98	30	01/02/2020
2	2940	98	30	01/03/2020
2	2940	98	30	01/04/2020
2	2940	98	30	01/05/2020
2	2940	98	30	01/06/2020
2	2940	98	30	01/07/2020
3	2940	70	42	01/02/2019
3	2940	70	42	01/03/2019
3	2940	70	42	01/04/2019
3	2940	70	42	01/05/2019
3	2940	70	42	01/06/2019
3	2940	70	42	01/07/2019
3	2940	70	42	01/08/2019
3	2940	70	42	01/09/2019
3	2940	70	42	01/10/2019
3	2940	70	42	01/11/2019
3	2940	70	42	01/12/2019
3	2940	70	42	01/01/2020
3	2940	70	42	01/02/2020
3	2940	70	42	01/03/2020
3	2940	70	42	01/04/2020
3	2940	70	42	01/05/2020
3	2940	70	42	01/06/2020
3	2940	70	42	01/07/2020

4	405	15	27	01/05/2019
4	1620	60	27	01/06/2019
4	1620	60	27	01/07/2019
4	1620	60	27	01/08/2019
4	1620	60	27	01/09/2019
4	1620	60	27	01/10/2019
4	1620	60	27	01/11/2019
4	1620	60	27	01/12/2019
5	640	20	32	01/06/2019
5	1600	50	32	01/07/2019
5	1600	50	32	01/09/2019
5	1600	50	32	01/08/2019
5	1600	50	32	01/10/2019
5	1600	50	32	01/11/2019
5	1280	40	32	01/12/2019
6	1160	40	29	01/08/2019
6	1160	40	29	01/09/2019
6	1160	40	29	01/10/2019
6	1160	40	29	01/11/2019
6	1160	40	29	01/12/2019
6	1160	40	29	01/01/2020
6	1160	40	29	01/03/2020
6	1160	40	29	01/02/2020
6	1160	40	29	01/04/2020
6	1160	40	29	01/05/2020
6	1160	40	29	01/06/2020
6	1160	40	29	01/07/2020
7	580	20	29	01/08/2019
7	1450	50	29	01/09/2019
7	1450	50	29	01/10/2019
7	1450	50	29	01/11/2019
7	1450	50	29	01/12/2019
7	203	7	29	01/02/2020
7	1450	50	29	01/01/2020
8	336	14	24	01/10/2019

8	840	35	24	01/11/2019
8	840	35	24	01/12/2019
8	840	35	24	01/01/2020
8	96	4	24	01/02/2020
9	1380	23	60	01/03/2020

Orders – table

last_contact	id_books	transaction_number	id_customers	status	date	id_orders
23/01/2018	12652	18763250	154	Completed	20/01/2018	23
22/05/2018	12649	18763251	155	Completed	20/05/2018	25
13/12/2018	12650	18763252	156	Completed	07/12/2018	26
01/10/2019	12656	18763253	157	Completed	25/09/2019	27
01/10/2019	12657	18763253	157	Completed	25/09/2019	28
25/11/2019	12658	18763254	158	Completed	20/11/2019	29
05/12/2019	12659	18763255	158	Completed	01/12/2019	30
12/01/2020	12660	18763256	159	Completed	08/01/2020	31
25/06/2020	12663	18763257	156	Waiting for Pickup	20/06/2020	32
28/07/2020	12661	18763258	157	Ordered	28/07/2020	33
28/07/2020	12662	18763259	158	Ordered	28/07/2020	34

Store payment- table

Other	rent	telephones	mobile_phones	property_tax	electricity	water	payment_date
700	2000	100	50	500	200	100	01/01/2018
700	2000	100	50	500	200	100	01/03/2018
200	2000	100	50	500	400	150	01/04/2018
200	2000	100	50	500	400	150	01/05/2018
378	2000	100	50	500	200	120	01/06/2018
0	2000	100	50	500	770	160	01/07/2018
100	2000	100	50	550	450	70	01/08/2018
500	2000	140	120	550	50	90	01/09/2018
600	2000	140	120	550	50	100	01/10/2018
200	2000	100	120	550	300	40	01/11/2018
300	2000	100	60	550	700	110	01/12/2018
200	2500	100	60	600	400	100	01/01/2019
130	2500	100	60	600	446	120	01/02/2019
200	2500	100	60	600	358	124	01/03/2019
400	2500	100	60	600	579	90	01/04/2019
460	2500	100	79	600	435	86	01/05/2019
170	2500	120	79	600	337	246	01/06/2019
370	2500	120	79	610	560	124	01/07/2019
8000	2500	120	79	610	350	100	01/08/2019
203	2500	120	79	610	350	100	01/09/2019
453	2500	120	79	600	440	110	01/10/2019
325	2500	120	79	600	190	140	01/11/2019
1000	2500	120	79	600	559	189	01/12/2019
700	2700	170	79	610	390	100	01/01/2020
829	2700	170	79	610	456	178	01/02/2020
20	2700	170	79	610	230	120	01/03/2020
110	1350	170	79	305	154	120	01/04/2020
0	1350	120	40	305	89	30	01/05/2020
120	1800	120	40	600	143	80	01/06/2020
240	2200	120	40	600	223	137	01/07/2020
456	2000	100	50	500	200	90	01/02/2018

Store purchase- table

origin	price	id_books	date	id_store_purchases
Client	10	12647	03/01/2018	1
Library	15	12648	22/05/2018	2
Library	25	12649	22/05/2018	3
Client	20	12650	12/12/2018	4
Client	30	12651	17/01/2019	5
Client	20	12652	03/02/2019	6
Library	30	12653	01/03/2019	7
Library	10	12654	16/04/2019	8
Client	45	12655	21/08/2019	9
Library	50	12656	30/09/2019	10
Library	20	12657	30/09/2019	11
Client	5	12658	24/11/2019	12
Client	30	12659	03/12/2019	13
Library	40	12660	12/01/2020	14
Client	40	12664	01/07/2020	15
Library	20	12665	02/08/2019	16
Client	22	12666	03/08/2019	17

Transaction – table

payment_type	id_delivery	total_price	delivery_price	books_price	total_weight	id_books	id_orders	id_customers	date	transaction_number	id_transactions
Bit	3	62.6	8.6	54	0.86	12652	23	154	20/01/2018	18763250	55
Credit Card	1	45	0	45	0.89	12649	25	155	22/05/2018	18763251	56
Bank Transfer	6	92.5	42.5	50	1.7	12650	26	156	07/12/2018	18763252	57
Credit Card	5	118.2	18.2	100	1.82	12656	27	157	25/09/2019	18763253	58
Credit Card	2	47.4	2.4	45	0.12	12657	28	157	25/09/2019	18763253	59

Bit	6	23.25	3.25	20	0.13	12658	29	158	20/11/2019	18763254	60
Bit	5	84.2	14.2	70	1.42	12659	30	158	01/12/2019	18763255	61
Bank Transfer	6	102.5	36.5	66	1.46	12660	31	159	08/01/2020	18763256	62
NULL	1	45	0	45	0.64	12663	32	156	20/06/2020	18763257	63
NULL	1	30	0	30	0.62	12661	33	157	28/07/2020	18763258	64
Bit	1	50	0	50	0.87	12662	34	158	28/07/2020	18763259	65
Credit Card	2	104	24	80	1.2	12665	NULL	156	15/08/2019	18763260	67
Credit Card	2	104	24	80	1.2	12666	NULL	156	15/08/2019	18763260	68

The code in SQL that implements the queries:

;'%SELECT title,location FROM bookstore.books WHERE title LIKE '%VAR .1

;Select * from bookstore.customers WHERE join_date LIKE (select MIN(join_date) from bookstore.customers) .2

;Select * from bookstore.books WHERE arrival_date LIKE (select MIN(arrival_date) from bookstore.books) .3

;SELECT * FROM bookstore.orders order by date .4

SELECT boo.title, count(boo.title) as times_bought from bookstore.transactions as tra INNER JOIN bookstore.books as boo ON boo.id_books = tra.id_books WHERE .5
boo.title LIKE '%HOB%' group by title

SELECT author_name, num_of_readings FROM (SELECT boo.author_name, count(boo.author_name) as num_of_readings, Rank() over (order by .6
count(boo.author_name) desc) as Rnk FROM bookstore.books as boo INNER JOIN bookstore.transactions as tra ON boo.id_books = tra.id_books WHERE tra.date between
'2018-01-01' and '2020-07-20' GROUP BY boo.author_name) as temp_result WHERE Rnk=1

SELECT cus.first_name,cus.last_name,count(cus.id_customers) as books_bought FROM bookstore.customers as cus INNER JOIN bookstore.transactions as tra ON .7
cus.id_customers = tra.id_customers GROUP BY cus.id_customers ORDER BY COUNT(books_bought) DESC LIMIT 3

SELECT title, count(title) as num_of_translations FROM (SELECT boo.title, boo.translator_name from bookstore.books as boo WHERE location = 'Store' AND .8
boo.translator_name NOT LIKE "NULL" OR location = 'Warehouse' AND boo.translator_name NOT LIKE "NULL" GROUP BY translator_name ORDER BY
COUNT(translator_name)) as temp_result GROUP BY title ORDER BY count(title) DESC LIMIT 1

```
SELECT  books.title,transactions.date,transactions.total_price  FROM bookstore.transactions INNER JOIN bookstore.books ON  .9
bookstore.transactions.id_books=bookstore.books.id_books WHERE id_customers in (SELECT id_customers FROM bookstore.customers WHERE first_name = "Mor" AND
;last_name = "Tahar" )
```

```
SELECT books.title, orders.date, orders.status, transactions.payment_status FROM bookstore.orders INNER JOIN bookstore.books ON  .10
bookstore.orders.id_books=bookstore.books.id_books INNER JOIN bookstore.transactions ON
bookstore.orders.transaction_number=bookstore.transactions.transaction_number WHERE orders.id_customers in (select id_customers FROM bookstore.customers
;WHERE first_name = "Nirush" AND last_name = "Hazanush") ORDER BY orders.date
```

.11

```
SELECT id_books,title FROM bookstore.books; [INTERACTIVE-OPTIONAL]
```

```
SELECT * FROM bookstore.delivery; [INTERACTIVE-OPTIONAL]
```

```
SELECT ((SELECT ROUND(sum(weight),2) as total_weight FROM bookstore.books WHERE id_books in ('12647','12648')) * (SELECT price_per_kg FROM bookstore.delivery
;WHERE id_delivery = "4")) as calculated_price
```

```
SELECT books.title, tb1.total_price, delivery.company, delivery.delivery_type, tb1.shipment_address, tb1.tracking_number, tb1.delivery_status,  .12
tb1.transaction_number  FROM bookstore.transactions tb1 INNER JOIN bookstore.books ON tb1.id_books=bookstore.books.id_books INNER JOIN bookstore.delivery ON
tb1.id_delivery=bookstore.delivery.id_delivery JOIN bookstore.transactions tb2 ON tb1.transaction_number=tb2.transaction_number AND tb1.id_books != tb2.id_books
AND tb1.id_delivery != tb2.id_delivery OR tb1.transaction_number=tb2.transaction_number AND tb1.id_books != tb2.id_books AND tb1.shipment_address !=
;tb2.shipment_address WHERE tb1.id_customers in (SELECT id_customers FROM bookstore.customers WHERE first_name = "Liran" AND last_name = "Dafadi")
```

```
SELECT customers.first_name,customers.last_name,books.title,transactions.delivery_status FROM bookstore.transactions INNER JOIN bookstore.books ON  .13
bookstore.books.id_books=bookstore.transactions.id_books INNER JOIN bookstore.customers ON
;'%bookstore.transactions.id_customers=bookstore.customers.id_customers WHERE transactions.id_transactions = '%VARID
```

```
;(5,6) SELECT SUM(delivery_price) AS Xpress_Delivery_Price FROM bookstore.transactions WHERE YEAR(date) = 2019 AND MONTH(date) = 11 AND id_delivery in .14
```

```
;"SELECT SUM(total_price) AS BitPayment FROM bookstore.transactions WHERE YEAR(date) = 2019 AND MONTH(date) = 11 AND payment_type = "Bit .15
```

```
SELECT * FROM bookstore.transactions WHERE books_price > (Select AVG(books_price) From bookstore.transactions WHERE date >= DATE_SUB(NOW(),INTERVAL 1  .16
;YEAR)) AND date >= DATE_SUB(NOW(),INTERVAL 1 YEAR)
```

```
Select SUM(id_delivery in (2,3,4)) AS 'Israel Post', SUM(id_delivery in (5,6)) AS 'Xpress' FROM bookstore.transactions WHERE date >= DATE_SUB(NOW(),INTERVAL 1  .17
;YEAR)
```

```
SELECT boo1.title,boo1.id_books,boo1.translator_name,boo1.publisher_name,boo1.year_of_distribution,tra.total_price, tra.shipment_address,  .18
tra.tracking_number,tra.delivery_status, tra.transaction_number FROM bookstore.transactions as tra JOIN bookstore.books boo1 ON tra.id_books = boo1.id_books  JOIN
bookstore.books boo2 ON boo1.title=boo2.title AND boo1.translator_name != boo2.translator_name OR boo1.title=boo2.title AND boo1.publisher_name !=
boo2.publisher_name OR boo1.title=boo2.title AND boo1.year_of_distribution != boo2.year_of_distribution WHERE tra.id_delivery !=1 AND transaction_number in (select
transaction_number from bookstore.transactions group by transaction_number having count(*) > 1)
```

```
SELECT tb1.date as last_purchase_date,cs.first_name,cs.last_name,cs.mobile_number from transactions tb1 INNER JOIN bookstore.customers cs ON  .19
tb1.id_customers=cs.id_customers where tb1.payment_status = 'Payed' AND tb1.date < DATE_SUB(NOW(), INTERVAL 2 YEAR) AND tb1.id_customers not in (select
id_customers from bookstore.transactions where date > DATE_SUB(NOW(), INTERVAL 2 YEAR) group by id_customers)
```

```
SELECT cus.first_name,cus.last_name,cus.mobile_number,ord.id_orders,ord.last_contact,boo.title,boo.location,tra.payment_status FROM bookstore.orders ord INNER  .20
JOIN bookstore.transactions tra ON ord.id_orders = tra.id_orders AND tra.payment_status = 'Not Payed' INNER JOIN bookstore.books boo ON ord.id_books = boo.id_books
INNER JOIN bookstore.customers cus ON ord.id_customers = cus.id_customers WHERE ord.last_contact < DATE_SUB(NOW(),  ('ןStore','Warehouse') AND boo.location in
; 'INTERVAL 14 DAY) AND ord.status = 'Waiting For Pickup
```

```
SELECT payment_date, @b := cc+@b as books_in_warehouse FROM (SELECT sto.payment_date, count(boo.location) as cc, @b:=0 as var from  .21
bookstore.store_payments as sto LEFT JOIN (SELECT * FROM bookstore.books WHERE location = 'warehouse') as boo on YEAR(boo.arrival_date) LIKE
YEAR(sto.payment_date) AND MONTH(boo.arrival_date) LIKE MONTH(sto.payment_date)  group by sto.payment_date,location order by sto.payment_date) as temp_table
```

```
SELECT count(id_store_purchases)  as books_bought , sum(price) as price_payed FROM bookstore.store_purchases WHERE date between "2018-01-01" and "2019-05- .22
;"23
```

```
select (SELECT sum(tra.total_price) FROM bookstore.transactions tra where year(tra.date) = 2019 and month(tra.date) = 09) - (SELECT sum(pur.price) FROM  .23
bookstore.store_purchases pur where year(pur.date) = 2019 and month(pur.date) = 09) as profit
```

```
SELECT YEAR(payment_date) as year, count(tra.id_transactions) as transaction_count, ROUND(count(tra.id_transactions)/12,3) as yearly_avrage_monthly_cut from  .24
bookstore.store_payments as sto LEFT JOIN (SELECT * FROM bookstore.transactions) as tra on YEAR(tra.date) LIKE YEAR(sto.payment_date) and MONTH(tra.date) LIKE
MONTH(sto.payment_date) group by YEAR(sto.payment_date)
```

```
SELECT emp.full_name, sal.salary_sum, sal.salary_date FROM bookstore.employees_salary sal INNER JOIN bookstore.employees emp ON  .25
;'sal.id_employees=emp.id_employees WHERE month(sal.salary_date) = '05' AND year(sal.salary_date) = '2020' AND emp.full_name = 'Dana Orr
```

```
SELECT full_name, books_sold FROM (SELECT emp.full_name,count(tra.id_employees) as books_sold, Rank() over (order by count(tra.id_employees)) as rnk FROM  .26
bookstore.transactions tra INNER JOIN bookstore.employees emp ON tra.id_employees=emp.id_employees WHERE year(tra.date) = '2018' AND month(tra.date) = '05'
;group by tra.id_employees order by count(tra.id_employees)) as temp_table WHERE rnk=1
```