

**מיני פרויקט
בבסיסי נתונים**

ניהול מסעדה

מגישה : שירן שלומי



שלב א'

משימה 1

- 1.1 אפיון המערכת
- 2.1 רשימת ישויות
- 3.1 תרשים ישויות ERD
- 4.1 תרשים סכמות DSD
- 5.1 יצירת הטבלאות פקודת DDL

משימה 2

- 2.1 פקודת INSERT
- 2.2 פקודת DROP
- 2.3 פקודת SELECT

משימה 3

- 3.1 שאילתות SELECT
- 3.2 שאילתות UPDATE
- 3.3 שאילתות DELETE

משימה 4

VIEWS

משימה 5

- 5.1 שאילתות עם פרמטרים
- 5.2 הכנסת נתונים ע"י DATE GENERATOR

משימה 6

- 6.1 יבוא נתונים מקובץ
- 6.2 גיבוי ושחזור נתונים

שלב ב'

משימה 7

7.1 שימוש בפרויקט

7.2 דוחות

7.3 גרפים

משימה 8

ALTER פקודת

משימה 9

9.1 הרשאת גישה GRANT וביטול הרשאת גישה REVOKE

9.2 פקודות COMMIT ו- ROLLBACK

משימה 10

פרוצדורות

משימה 11

פונקציות

משימה 12

טריגרים

שלב ג'

אינטגרציה

שלב א'

1.1 אפיון המערכת

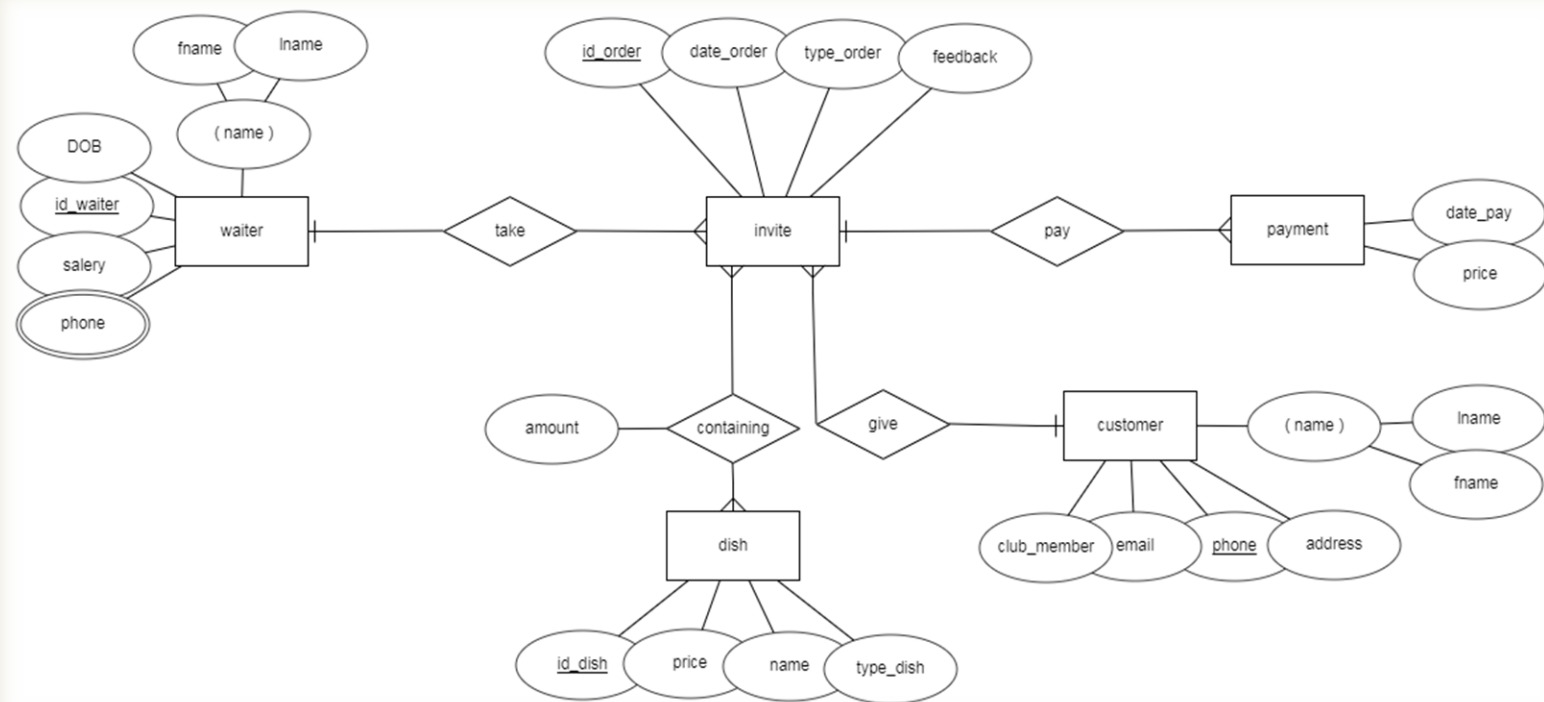
המערכת הינה מערכת לניהול בסיס נתונים עבור מסעדה.
במערכת קיימים עובדים-מלצרים שעליהם לקחת הזמנות מלקוחות.
כל הזמנה מכילה מנות בהתאם שהוזמנו על ידי הלקוח .
עבור כל הזמנה יש תשלום.

1.2 רשימת ישויות

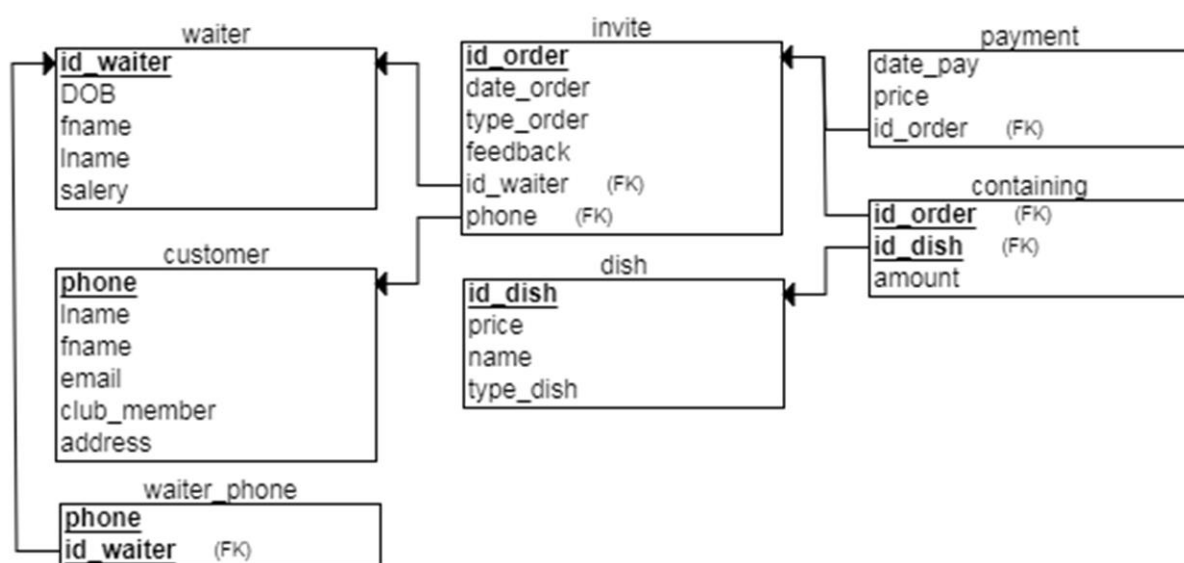
רשימת הישויות בהן עוסקת המערכת:

מלצרים
לקוחות
מנות
הזמנות
תשלום

1.3 תרשים ERD



1.4 תרשים DSD



1.5 יצירת הטבלאות – פקודת create

SQL	Output	Statistics
-----	--------	------------

```
CREATE TABLE waiter
(
    id_waiter VARCHAR2(10) NOT NULL,
    DOB DATE NOT NULL,
    fname VARCHAR2(20) NOT NULL,
    lname VARCHAR2(20) NOT NULL,
    salary NUMBER(10) NOT NULL,
    PRIMARY KEY (id_waiter)
);

CREATE TABLE customer
(
    lname VARCHAR2(20) NOT NULL,
    fname VARCHAR2(20) NOT NULL,
    phone NUMBER(10) NOT NULL,
    email VARCHAR2(20) NOT NULL,
    club_member NUMBER(10) NOT NULL,
    address VARCHAR2(20) NOT NULL,
    PRIMARY KEY (phone)
);

CREATE TABLE dish
(
    id_dish VARCHAR2(10) NOT NULL,
    price NUMBER(10) NOT NULL,
    name VARCHAR2(20) NOT NULL,
    type_dish VARCHAR(20) NOT NULL,
    PRIMARY KEY (id_dish)
);

CREATE TABLE waiter_phone
(
    phone NUMBER(10) NOT NULL,
    id_waiter VARCHAR2(10) NOT NULL,
    PRIMARY KEY (phone),
    FOREIGN KEY (id_waiter) REFERENCES waiter(id_waiter)
);
```

```
CREATE TABLE invite
(
    id_order VARCHAR2(10) NOT NULL,
    date_order DATE NOT NULL,
    type_order VARCHAR2(20) NOT NULL,
    feedback VARCHAR2(20) NOT NULL,
    id_waiter VARCHAR2(10) NOT NULL,
    phone NUMBER(10) NOT NULL,
    PRIMARY KEY (id_order),
    FOREIGN KEY (id_waiter) REFERENCES waiter(id_waiter),
    FOREIGN KEY (phone) REFERENCES customer(phone)
);

CREATE TABLE payment
(
    date_pay DATE NOT NULL,
    price NUMBER(10) NOT NULL,
    id_order VARCHAR2(10) NOT NULL,
    FOREIGN KEY (id_order) REFERENCES invite(id_order)
);

CREATE TABLE containing
(
    amount NUMBER (20) NOT NULL,
    id_order VARCHAR2(10) NOT NULL,
    id_dish VARCHAR2(10) NOT NULL,
    PRIMARY KEY (id_order, id_dish),
    FOREIGN KEY (id_order) REFERENCES invite(id_order),
    FOREIGN KEY (id_dish) REFERENCES dish(id_dish)
);
```

2.1 פקודת INSERT

Insert to waiter

SQL	Output	Statistics
<pre> insert into waiter (id_waiter, DOB, fname, lname, salary) values(203432117,to_date('01-02-1997','dd-mm-yyyy'),'shir','levi',9000); insert into waiter (id_waiter, DOB, fname, lname, salary) values(433432117,to_date('02-08-1987','dd-mm-yyyy'),'elroyi','cohen',4900); insert into waiter (id_waiter, DOB, fname, lname, salary) values(243432817,to_date('08-08-1996','dd-mm-yyyy'),'tamar','somech',6600); insert into waiter (id_waiter, DOB, fname, lname, salary) values(261792117,to_date('09-05-1990','dd-mm-yyyy'),'lea','levi',6500); insert into waiter (id_waiter, DOB, fname, lname, salary) values(203487217,to_date('07-06-2000','dd-mm-yyyy'),'moti','dan',7000); insert into waiter (id_waiter, DOB, fname, lname, salary) values(303432117,to_date('05-11-2002','dd-mm-yyyy'),'miri','haron',5690); insert into waiter (id_waiter, DOB, fname, lname, salary) values(203436948,to_date('08-10-1993','dd-mm-yyyy'),'nir','levi',5500); insert into waiter (id_waiter, DOB, fname, lname, salary) values(947232117,to_date('18-05-1998','dd-mm-yyyy'),'shira','noach',6000); insert into waiter (id_waiter, DOB, fname, lname, salary) values(264987515,to_date('10-12-2005','dd-mm-yyyy'),'moshe','cohen',9100); insert into waiter (id_waiter, DOB, fname, lname, salary) values(216887584,to_date('10-05-2010','dd-mm-yyyy'),'dor','dadon',8000); </pre>		

Insert to customer

SQL	Output	Statistics
<pre> insert into customer (lname, fname, phone, email, club_member, address) values('shalom','michal',0506561604,'mic@gmail.com',898,'89 derech amelech'); insert into customer (lname, fname, phone, email, club_member, address) values('cohen','yonatan',0508496284,'yoni8@walla.com',54,'62 menachem begin'); insert into customer (lname, fname, phone, email, club_member, address) values('kadosh','yoel',0507496281,'yo228@walla.com',162,'5 yoni nethaniho'); insert into customer (lname, fname, phone, email, club_member, address) values('bloch','miri',0529456118,'miri@gmail.com',1187,'9 apalmach'); insert into customer (lname, fname, phone, email, club_member, address) values('cocavi','ela',0504592687,'el111la@walla.com',282,'62 ben guryon'); insert into customer (lname, fname, phone, email, club_member, address) values('levi','yonatan',0508126483,'yonlev@gmail.com',47,'1 simcha holtberg'); insert into customer (lname, fname, phone, email, club_member, address) values('cohen','rona',0524165985,'ronacohen@walla.com',124,'1 menachem begin'); insert into customer (lname, fname, phone, email, club_member, address) values('amitay','gali',0504168272,'galush@gmail.com',941,'81 asivim'); commit; insert into customer (lname, fname, phone, email, club_member, address) values('shem-tov','nir',0541652973,'nirshemtov@gmail.com',601,'2 arishonim'); insert into customer (lname, fname, phone, email, club_member, address) values('hadad','noa',0501659426,'noa11h@walla.com',578,'22 harava'); commit; </pre>		

Insert to dish

SQL	Output	Statistics
<pre>insert into dish (id_dish, price, name, type_dish) values(1, 45, 'pasta', 'main dish'); insert into dish (id_dish, price, name, type_dish) values(2, 40, 'pizza', 'main dish'); insert into dish (id_dish, price, name, type_dish) values(3, 52, 'fish and chips', 'main dish'); insert into dish (id_dish, price, name, type_dish) values(4, 45, 'salad', 'main dish'); insert into dish (id_dish, price, name, type_dish) values(5, 31, 'soup', 'first dish'); insert into dish (id_dish, price, name, type_dish) values(6, 20, 'cheese bread', 'first dish'); insert into dish (id_dish, price, name, type_dish) values(7, 27, 'cheesecake', 'last dish'); insert into dish (id_dish, price, name, type_dish) values(8, 30, 'apple pie', 'last dish'); insert into dish (id_dish, price, name, type_dish) values(9, 10, 'ice cream', 'last dish'); insert into dish (id_dish, price, name, type_dish) values(10, 25, 'chocolate cake', 'last dish');</pre>		

Insert to waiter phone

SQL	Output	Statistics
<pre>insert into waiter_phone (phone, id_waiter) values(0506561602, 203432117); insert into waiter_phone (phone, id_waiter) values(0526495812, 433432117); insert into waiter_phone (phone, id_waiter) values(0523694587, 243432817); insert into waiter_phone (phone, id_waiter) values(054985164, 261792117); insert into waiter_phone (phone, id_waiter) values(0549852362, 203487217); insert into waiter_phone (phone, id_waiter) values(0521462587, 303432117); insert into waiter_phone (phone, id_waiter) values(052316492, 203436948); insert into waiter_phone (phone, id_waiter) values(0506249855, 947232117); insert into waiter_phone (phone, id_waiter) values(0503659954, 264987515); insert into waiter_phone (phone, id_waiter) values(0549659999, 216887584);</pre>		

Insert to invite

SQL	Output	Statistics
<pre>insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone) values(1, to_date('01-04-2020','dd-mm-yyyy'), 'sitting', 5, 203432117, 0506561604); insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone) values(2, to_date('01-04-2020','dd-mm-yyyy'), 'sitting', 5, 433432117, 0508496284); insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone) values(3, to_date('02-04-2020','dd-mm-yyyy'), 'sitting', 4, 243432817, 0507496281); insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone) values(4, to_date('11-02-2020','dd-mm-yyyy'), 'sitting', 4, 261792117, 0529456118); insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone) values(5, to_date('08-01-2020','dd-mm-yyyy'), 'sitting', 3, 203487217, 0504168272); insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone) values(6, to_date('01-01-2020','dd-mm-yyyy'), 'take away', 3, 303432117, 0524165985); insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone) values(7, to_date('10-02-2020','dd-mm-yyyy'), 'take away', 2, 203436948, 0508126483); insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone) values(8, to_date('03-03-2020','dd-mm-yyyy'), 'take away', 1, 947232117, 0504592687); insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone) values(9, to_date('20-04-2020','dd-mm-yyyy'), 'take away', 5, 264987515, 0541652973); insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone) values(10, to_date('15-01-2020','dd-mm-yyyy'), 'take away', 5, 216887584, 0501659426); insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone) values(11, to_date('10-02-2020','dd-mm-yyyy'), 'take away', 2, 203436948, 0508126483); insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone)</pre>		

Insert to payment

SQL	Output	Statistics
<pre>insert into payment (date_pay, price, id_order) values(to_date('08-01-2020','dd-mm-yyyy'),250 , 5); insert into payment (date_pay, price, id_order) values(to_date('02-04-2020','dd-mm-yyyy'),250 , 3); insert into payment (date_pay, price, id_order) values(to_date('01-04-2020','dd-mm-yyyy'),150 , 2); insert into payment (date_pay, price, id_order) values(to_date('01-04-2020','dd-mm-yyyy'),300 , 1); insert into payment (date_pay, price, id_order) values(to_date('11-02-2020','dd-mm-yyyy'),320 , 4); insert into payment (date_pay, price, id_order) values(to_date('01-01-2020','dd-mm-yyyy'),80 , 6); insert into payment (date_pay, price, id_order) values(to_date('01-02-2020','dd-mm-yyyy'),409 , 7); insert into payment (date_pay, price, id_order) values(to_date('20-04-2020','dd-mm-yyyy'),420 , 9); insert into payment (date_pay, price, id_order) values(to_date('01-01-2020','dd-mm-yyyy'),275 , 10); insert into payment (date_pay, price, id_order) values(to_date('10-02-2020','dd-mm-yyyy'),180 , 11); insert into payment (date_pay, price, id_order)</pre>		

Insert to containing

SQL	Output	Statistics
<pre>insert into containing (amount, id_order, id_dish) values(1, 2, 2); insert into containing (amount, id_order, id_dish) values(3, 2, 1); insert into containing (amount, id_order, id_dish) values(1, 3, 3); insert into containing (amount, id_order, id_dish) values(2, 4, 5); insert into containing (amount, id_order, id_dish) values(5, 5, 4); insert into containing (amount, id_order, id_dish) values(4, 10, 6); insert into containing (amount, id_order, id_dish) values(2, 7, 8); insert into containing (amount, id_order, id_dish) values(1, 8, 9); insert into containing (amount, id_order, id_dish) values(3, 9, 10); insert into containing (amount, id_order, id_dish) values(2, 6, 2); insert into containing (amount, id_order, id_dish) values(1, 8, 9);</pre>		

2.2 מחיקת הטבלאות – פקודת DROP

```
SQL      Output  Statistics
drop table containing ;
drop table payment ;
drop table invite ;
drop table waiter_phone ;
drop table dish ;
drop table customer ;
drop table waiter;
```

2.3 הצגת/בחירת הטבלאות - פקודת SELECT

Select waiter table

SQL

Output

Statistics

select * from waiter
commit;

	ID_WAITER	DOB	FNAME	LNAME	SALARY
▶ 1	203432117	01/02/1997	shir	levi	9000
2	433432117	02/08/1987	elroyi	cohen	4900
3	243432817	08/08/1996	tamar	somech	6600
4	261792117	09/05/1990	lea	levi	6500
5	203487217	07/06/2000	moti	dan	7000
6	303432117	05/11/2002	miri	haron	5690
7	203436948	08/10/1993	nir	levi	5500
8	947232117	18/05/1998	shira	noach	6000
9	264987515	10/12/2005	moshe	cohen	9100

Select customer table


SQL

Output

Statistics

select * from customer

commit;



	LNAME	FNAME	PHONE	EMAIL	CLUB_MEMBER	ADDRESS			
1	shalom	506561604	mic@gmail.com	...	898	89 derech amelech	...
2	cohen	508496284	yonis8@walla.com	...	54	62 menachem begin	...
3	kadosh	507496281	yo228@walla.com	...	162	5 yoni nethaniho	...
4	bloch	529456118	miri@gmail.com	...	1187	9 apalmach	...
5	cocavi	504592687	elilla@walla.com	...	282	62 ben guryon	...
6	levi	508126483	yonlev@gmail.com	...	47	1 simcha holtberg	...
7	cohen	524165985	ronacohen@walla.com	...	124	1 menachem begin	...
8	amitay	504168272	galush@gmail.com	...	941	81 asivim	...
9	shem-tov	541652973	nirshemtov@gmail.com	...	601	2 arishonim	...

Select invite table


SQL

Output

Statistics

select * from invite

commit;



	ID_ORDER	DATE_ORDER	TYPE_ORDER	FEEDBACK	ID_WAITER	PHONE
1	1	01/04/2020	sitting	5	203432117	506561604
2	2	01/04/2020	sitting	5	433432117	508496284
3	3	02/04/2020	sitting	4	243432817	507496281
4	4	11/02/2020	sitting	4	261792117	529456118
5	5	08/01/2020	sitting	3	203487217	504168272
6	6	01/01/2020	take away	3	303432117	524165985
7	7	10/02/2020	take away	2	203436948	508126483
8	8	03/03/2020	take away	1	947232117	504592687
9	9	20/04/2020	take away	5	264987515	541652973
10	10	15/01/2020	take away	5	216887584	501659426
11	11	10/02/2020	take away	2	203436948	508126483
12	12	03/03/2020	take away	1	947232117	504592687
13	13	20/04/2020	take away	5	264987515	541652973
14	14	15/01/2020	take away	5	216887584	504592687


Select dish table

SQL

Output


Statistics

```
select * from dish
commit;
|
```



	ID_DISH	PRICE	NAME	TYPE_DISH
3	3	52	fish and chips	main dish
4	4	45	salad	main dish
5	5	31	soup	first dish
6	6	20	cheese bread	first dish
7	7	27	cheesecake	last dish
8	8	30	apple pie	last dish
9	9	10	ice cream	last dish
10	10	25	chocolate cake	last dish

Select waiter_phone table

SQL	Output	Statistics
<pre>select * from waiter_phone commit;</pre>		
		
	PHONE	ID_WAITER
1	506561602	203432117
2	526495812	433432117
3	523694587	243432817
4	54985164	261792117
5	549852362	203487217
6	521462587	303432117
7	52316492	203436948
8	506249855	947232117













Select payment table

SQL

Output

Statistics

```
select * from payment
commit;
```

		DATE_PAY	PRICE	ID_ORDER
▶	1	08/01/2020	225	5
	2	02/04/2020	52	3
	3	01/04/2020	175	2
	4	01/04/2020	160	1
	5	11/02/2020	124	4
	6	01/01/2020	260	6
	7	20/04/2020	225	9
	8	01/01/2020	80	10
	9	10/02/2020	340	11
	10	03/03/2020	60	12
	11	15/01/2020	160	14
	12	01/04/2020	135	16
	13	03/03/2020	117	8

Select containing table

SQL

Output

Statistics

```
select * from containing
commit;
```

	AMOUNT	ID_ORDER	ID_DISH
2	3	2	1
3	1	3	3
4	2	4	5
5	5	5	4
6	4	10	6
7	2	7	8
8	1	8	9
9	3	9	10
10	2	6	2
11	3	9	6
12	4	6	1
13	1	8	3
14	3	9	8
15	3	12	6
16	4	11	1
17	1	16	3
18	3	16	8
19	4	11	2
20	4	14	2


משימה 3

3.1 שאלות

שאלה 1

SQL Output Statistics

```
--1--  
-- העג עביר שנת 2020 את חס' ההזמנות  
  
select count(id_order) as "total order"  
from invite  
where date_order between to_date('01-01-2020','dd-mm-yyyy')  
and to_date('31-12-2020','dd-mm-yyyy')
```




	total order
1	16

שאלה 2

SQL Output Statistics

```
--2--  
-- העגת סכום המשכורות לחודש אפריל  
  
select sum(salary) as "April salary"  
from waiter  
where id_waiter in (select id_waiter from invite  
where date_order between to_date('01-04-2020','dd-mm-yyyy')  
and to_date('30-04-2020','dd-mm-yyyy'))
```



	April salary
1	37600

שאלתה3

SQL Output Statistics

```
--3--
--הצגת פרטי לקוח ששילם מעל 100 שקלים עבור הזמנה--
--השאלתה מחזירה טלפון ושם לקוח שעשה הזמנה שסכומה עולה על 100 שקלים וכן את הסכום ששילם--

select invite.id_order , price ,customer.phone , lname, fname
from invite join customer
on invite.phone = customer.phone
      join payment
on invite.id_order = payment.id_order
where (price>100)
order by price
|
```

	ID_ORDER	PRICE	PHONE	LNAME	FNAME
1	4	124	529456118	bloch	miri
2	16	135	508496284	cohen	yonatan
3	17	155	504592687	cocavi	ela
4	1	160	506561604	shalom	micchal
5	14	160	504592687	cocavi	ela
6	2	175	508496284	cohen	yonatan
7	9	225	541652973	shem-tov	nir
8	5	225	504168272	amitay	gali
9	6	260	524165985	cohen	rona
10	11	340	508126483	levi	yonatan

שאלתה4

SQL Output Statistics

```
--4--
--עובדים שביצעו יותר מהזמנה אחת--
--השאלתה תציג את שם העובד , מס' ההזמנות שהעובד לקח וכן את התאריך האחרון שהעובד ביצע הזמנה--

select lname,
       fname,
       waiter.id_waiter,
       count(id_order) as "total order",
       max(date_order) as "last order"
from waiter join invite
on waiter.id_waiter = invite.id_waiter
group by lname , fname ,waiter.id_waiter
having (count(id_order)>1)
```

	LNAME	FNAME	ID_WAITER	total order	last order
1	levi	shir	203432117	2	05/04/2020
2	levi	nir	203436948	2	10/02/2020
3	noach	shira	947232117	2	03/03/2020
4	cohen	moshe	264987515	2	20/04/2020
5	dadon	dor	216887584	4	01/04/2020

שאילתה 5

SQL Output Statistics

--5--
-- הזמנה שמחזירה גבוה ממאה שקלים היא רגועית
-- השאילתה מחזירה את מספר ההזמנות שמחזירן מעל 100 שקלים וכן-
-- את התשלום הכולל--

select count(invite.id_order) as "total order",
sum(price) as "total price"

from invite join payment
on invite.id_order = payment.id_order
where (price > 100) |

	total order	total price
▶ 1	10	1959

שאילתה 6

SQL Output Statistics

--6--
-- שאילתה שמחזירה את המחירים בסדר עולה של המנות הזולות יותר ממנה שמונה

select *
from dish
where price < (select price
from dish
where id_dish = 8)

order by price
|

	ID_DISH	PRICE	NAME	TYPE_DISH
▶ 1	9	10	ice creem	last dish
2	6	20	cheese bread	first dish
3	10	25	chocolate cake	last dish
4	7	27	cheesecake	last dish

שאלתה7

SQL Output Statistics

```
--7--
--השאלתה מחזירה מבין ההזמנות שבוצעו בחודש אפריל את המנה היקרה מבין המנות שהוזמנו באותה הזמנה--
--את סך המנות שהכילה הזמנה וכן את מס הזמנה--
--כל המנות שהוזמנו צריכות להיות מעל 20 שקלים--

select max(price)as "max price",
       count(dish.id_dish)as "total dishes" ,
       containing.id_order

from dish join containing
      on dish.id_dish = containing.id_dish
      join invite
      on containing.id_order = invite.id_order
where (date_order between to_date('01-04-2020','dd-mm-yyyy')
      and to_date('30-04-2020','dd-mm-yyyy'))

group by containing.id_order
having min(price) >20
```

	max price	total dishes	ID_ORDER
▶ 1	25	1	1
2	52	1	3
3	52	2	16
4	45	2	2

שאלתה8

SQL Output Statistics

```
--8--
--השאלתה מחזירה את מס ההזמנות עבור כל חודש שמחירן מעל 100 שח וכן היו הזמנות ישיבה--

select extract(month from date_order) as "Month",
       count(id_order) as "Num of Orders"
from invite
where id_order in (select id_order
                  from payment
                  where price>100)
and type_order='sitting'
group by extract(month from date_order)
order by "Num of Orders" desc
```

	Month	Num of Orders
▶ 1	4	3
2	7	1
3	1	1
4	2	1


שאלתה 9 (שאלתת בחינה)

SQL Output Statistics

--9--
השאלתה מחזירה את מס ההזמנות בטווח מחירים שהוגדרו מראש--
3 עמודות בהתאם--

select (select count(id_order)
from invite
where id_order in (select id_order
from payment
where price between 0 and 30)) as "price is 0-30",
(select count(id_order)
from invite
where id_order in (select id_order
from payment
where price between 30 and 100))as "price is 30-100",
(select count(id_order)
from invite
where id_order in (select id_order
from payment
where price between 100 and 300))as "price is 100-300"

from invite
where (rownum <2) |



	price is 0-30	price is 30-100	price is 100-300
▶ 1	0	4	9

3.1 שאלות UPDATE

שאלתה1

SQL Output Statistics

--1--
תוספת בונוס של 500 שקלים לעובדים שביצעו יותר מהזמנה אחת

update waiter
set salary = salary+500
where id_waiter in (select id_waiter
from invite
group by id_waiter
having (count(id_order)>1))

	ID_WAITER	DOB	FNAME	LNAME	SALARY
1	203432117	01/02/1997	shir	levi	9500
2	433432117	02/08/1987	elroyi	cohen	4900
3	243432817	08/08/1996	tamar	somech	6600
4	261792117	09/05/1990	lea	levi	6500
5	203487217	07/06/2000	moti	dan	7000
6	303432117	05/11/2002	miri	haron	5690
7	203436948	08/10/1993	nir	levi	6000
8	947232117	18/05/1998	shira	noach	6500
9	264987515	10/12/2005	moshe	cohen	9600
10	216887584	10/05/2010	dor	dadon	8500

שאלתה2

SQL Output Statistics

--2--
התיקרות של 10 שקלים עבור חנות שמחוצצת המכירה שלהן גדולה מ2 במועד אפריל--

update dish
set price = price+10
where id_dish in (select id_dish
from containing
where id_order in (select id_order
from invite
where (date_order between to_date('01-04-2020','dd-mm-yyyy')
and to_date('30-04-2020','dd-mm-yyyy'))
group by id_dish
having avg(amount)>2)

	ID_DISH	PRICE	NAME	TYPE_DISH
1	1	55	pasta	main dish
2	2	40	pizza	main dish
3	3	52	fish and chips	main dish
4	4	45	salad	main dish
5	5	31	soup	first dish
6	6	30	cheese bread	first dish
7	7	27	cheesecake	last dish
8	8	40	apple pie	last dish
9	9	10	ice cream	last dish
10	10	35	chocolate cake	last dish


3.3 שאלות DELETE

שאלה 1

SQL Output Statistics

--1--
-- השאלה מוזקת מתוך פרוט תכולת ההזמנה
-- את ההזמנה שמחירה גבוה מ-150 שקלים וכן הזמנה שמכילה את מנה 2
-- מתוך טבלת הזמנות / התפריט

delete
from containing
where id_order in (select id_order
 from payment
 where (price>150))
 and id_dish = 2 |



	AMOUNT	ID_ORDER	ID_DISH
▶ 1	3	2	1
2	1	3	3
3	2	4	5
4	5	5	4
5	4	10	6
6	2	7	8
7	1	8	9
8	3	9	10
9	3	9	6
10	4	6	1
11	1	8	3
12	3	9	8
13	3	12	6
14	4	11	1
15	1	16	3
16	3	16	8
17	2	1	10

שאלתה2

SQL

Output

Statistics

--2--

-- השאלתה מוחקת הזמנות לא רלוונטיות, ז"א הזמנות שלא חשונת 2020

-- וכן שמחירו עולה על 100 שח

```
delete
from invite
where id_order in (select id_order
                    from payment
                    where (date_pay between to_date('01-01-2019','dd-mm-yyyy')
                        and to_date('31-12-2019','dd-mm-yyyy'))
                    and payment.price>100)
```



	ID_ORDER	DATE_ORDER	TYPE_ORDER	FEEDBACK	ID_WAITER	PHONE
▶ 1	1	01/04/2020	sitting	5	203432117	506561604
2	2	01/04/2020	sitting	5	433432117	508496284
3	3	02/04/2020	sitting	4	243432817	507496281
4	4	11/02/2020	sitting	4	261792117	529456118
5	5	08/01/2020	sitting	3	203487217	504168272
6	6	01/01/2020	take away	3	303432117	524165985
7	7	10/02/2020	take away	2	203436948	508126483
8	8	03/03/2020	take away	1	947232117	504592687
9	9	20/04/2020	take away	5	264987515	541652973
10	10	15/01/2020	take away	5	216887584	501659426
11	11	10/02/2020	take away	2	203436948	508126483
12	12	03/03/2020	take away	1	947232117	504592687
13	13	20/04/2020	take away	5	264987515	541652973
14	14	15/01/2020	take away	5	216887584	504592687
15	15	05/04/2020	sitting	5	203432117	504592687
16	16	01/04/2020	sitting	5	216887584	508496284

משימה 4 – VIEW

1 view

הנהלת המסעדה החליטה לאזן את משכורות העובדים.
 ולכן החליטה שכל מלצר שמשתכר מעל 6000 יש לבדוק את יעילותו.
 לכן ניצור טבלה שתשמש את הנהלת המסעדה ובה רשימת כל העובדים שמשכורתם
 מעל 6000

טבלת ה-VIEW

```
SQL      Output  Statistics
create view salary_check as
select w.id_waiter,
       w.dob,
       w.fname,
       w.lname,
       w.salary,
       i.id_order,
       i.date_order,
       d.id_dish,
       d.price,
       d.name,
       d.type_dish,
       c.amount
from waiter w,invite i,dish d,containing c
where(salary>6000 and
       w.id_waiter=i.id_waiter and
       i.id_order=c.id_order and
       c.id_dish=d.id_dish)
order by salary
--
select * from salary_check
drop view salary_check
```

תוצאת הפקודה:

SQL

Output


Statistics

```
--
select * from salary_check
```

	ID_WAITER	DOB	FNAME	LNAME	SALARY	ID_ORDER	DATE_ORDER	ID_DISH	PRICE	NAME	TYPE_DISH	AMOUNT
1	261792117	09/05/1990	lea	levi	6500	4	11/02/2020	5	31	soup	first dish	2
2	243432817	08/08/1996	tamar	somech	6600	3	02/04/2020	3	52	fish and chips	main dish	1
3	203487217	07/06/2000	moti	dan	7000	5	08/01/2020	4	45	salad	main dish	5
4	216887584	10/05/2010	dor	dadon	8000	16	01/04/2020	3	52	fish and chips	main dish	1
5	216887584	10/05/2010	dor	dadon	8000	14	15/01/2020	2	40	pizza	main dish	4
6	216887584	10/05/2010	dor	dadon	8000	10	15/01/2020	6	20	cheese bread	first dish	4
7	216887584	10/05/2010	dor	dadon	8000	16	01/04/2020	8	30	apple pie	last dish	3
8	203432117	01/02/1997	shir	levi	9000	1	01/04/2020	10	25	chocolate cake	last dish	2
9	264987515	10/12/2005	moshe	cohen	9100	9	20/04/2020	8	30	apple pie	last dish	3
10	264987515	10/12/2005	moshe	cohen	9100	9	20/04/2020	10	25	chocolate cake	last dish	3
11	264987515	10/12/2005	moshe	cohen	9100	9	20/04/2020	6	20	cheese bread	first dish	3

שאיילתה 1 על ה-VIEW

הנהלת המסעדה רוצה לבדוק שאכן מלצר שמשתכר מעל 6000, לא מבצע פחות מ-2 הזמנות. (ז"א השאיילתה בודקת מתוך טבלת view איזה מלצר מבצע פחות מ-2 הזמנות בחודש)

SQL	Output	Statistics
<pre>-- השאיילתה בודקת האם כל מלצר שמשתכר מעל 6000 מבצע לפחות 2 הזמנות</pre> <pre>select id_waiter, extract(month from date_order) as "month", count(id_order) as "total orders" from salary_check sc group by extract(month from date_order) ,id_waiter having (count(id_order)<2)</pre>		
		
ID_WAITER	month	total orders
1 203432117	4	1
2 203487217	1	1
3 243432817	4	1
4 261792117	2	1

שאיילתה 2 על ה-VIEW

הנהלת המסעדה רוצה לבדוק שאכן מלצר שמשתכר מעל 6000, דואג להמליץ על מנות שמחירן ≤ 40 שקלים.

SQL



Output





Statistics






```
-- השאילתה מחזירה אלו מלצרים דואגים "לדחוף" ללקוחות את המנות
-- שמחירן לפחות 40 ש"ח








select id_waiter,id_order,id_dish,price
from salary_check sc
where price>=40

|
```







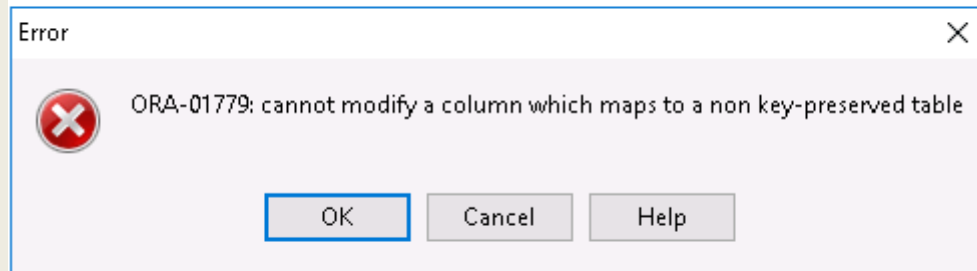


	ID_WAITER	ID_ORDER	ID_DISH	PRICE
▶ 1	243432817	3	3	52
2	203487217	5	4	45
3	216887584	16	3	52
4	216887584	14	2	40

עדכון נתונים על ה-VIEW

מלצרים שהשתכרו מעל 6000 אך לא עמדו בדרישות ההנהלה קוצצה משכורתם.

```
update salary_check sc
set salary=6000
where sc.id_waiter='203432117'
```



העדכון לא התבצע. לא ניתן לעדכן בהשקפה מכיוון שהנתונים תלויים בטבלאות אחרות.

מחיקת נתונים מה-VIEW

בעקבות ההחלטה לקצץ בשכר העובדים שלא עומדים בדרישות, לא היה מקובל על עובד בעל ת.ז. 203432117 ולכן ביקש להתפטר.

SQL

Output

Statistics

delete from salary_check sc
where sc.id_waiter='203432117'

	ID_WAITER	DOB	FNAME	LNAME	SALARY	ID_ORDER	DATE_ORDER	ID_DISH	PRICE	NAME	TYPE_DISH	AMOUNT
1	261792117	09/05/1990	lea	levi	6500	4	11/02/2020	5	31	soup	first dish	2
2	243432817	08/08/1996	tamar	somech	6600	3	02/04/2020	3	52	fish and chips	main dish	1
3	203487217	07/06/2000	moti	dan	7000	5	08/01/2020	4	45	salad	main dish	5
4	216887584	10/05/2010	dor	dadon	8000	16	01/04/2020	8	30	apple pie	last dish	3
5	216887584	10/05/2010	dor	dadon	8000	16	01/04/2020	3	52	fish and chips	main dish	1
6	216887584	10/05/2010	dor	dadon	8000	14	15/01/2020	2	40	pizza	main dish	4
7	216887584	10/05/2010	dor	dadon	8000	10	15/01/2020	6	20	cheese bread	first dish	4
8	264987515	10/12/2005	moshe	cohen	9100	9	20/04/2020	8	30	apple pie	last dish	3
9	264987515	10/12/2005	moshe	cohen	9100	9	20/04/2020	10	25	chocolate cake	last dish	3
10	264987515	10/12/2005	moshe	cohen	9100	9	20/04/2020	6	20	cheese bread	first dish	3

ההשקפה לאחר המחיקה. ניתן לראות שהמלצר בעל ת.ז. הנ"ל נמחק מההשקפה.

2 View

פעם בחודש הנהלת המסעדה מבצעת בדיקה בכל הקשור להוצאות והכנסות של המסעדה כדי לשמור על רווחיות העסק.

לכן ניצור טבלה שתשמש את הנהלת המסעדה ובה פרוט כל הוצאה/הכנסה

טבלת ה-VIEW

```
SQL Output Statistics

create view profit_check as
select w.id_waiter,
       w.dob,
       w.fname,
       w.lname,
       w.salary,
       i.id_order,
       i.date_order,
       i.type_order,
       i.feedback,
       i.phone,
       p.date_pay,
       p.price
from waiter w, invite i, payment p
where w.id_waiter=i.id_waiter
and   i.id_order=p.id_order

--

select * from profit_check
drop view profit_check
```


תוצאת הפקודה:

SQL

Output

Statistics

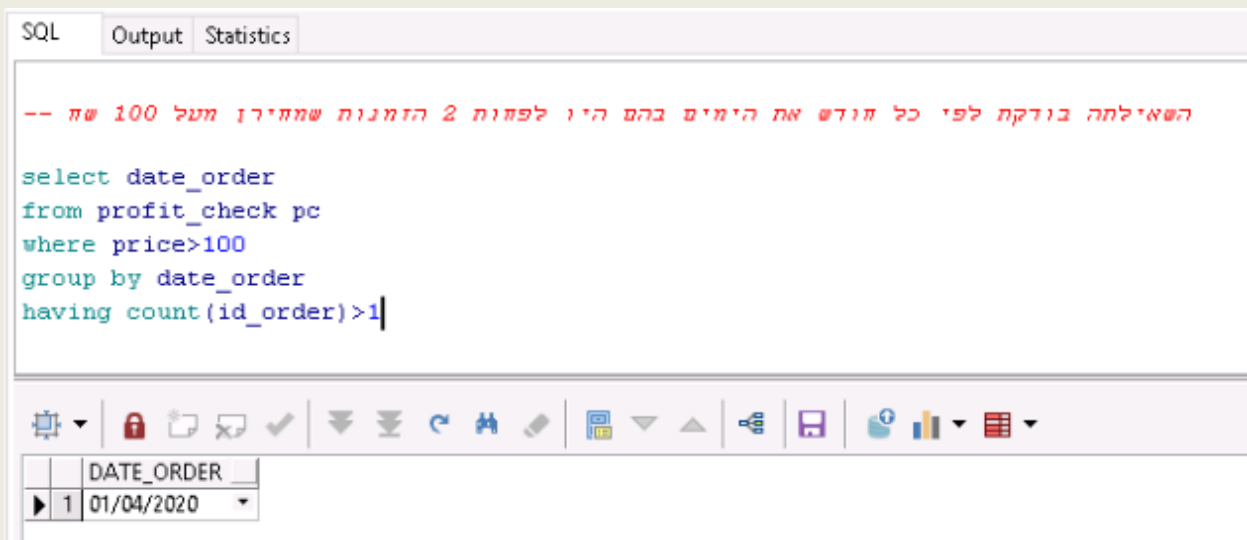
select * from profit_check



	ID_WAITER	DOB	FNAME	LNAME	SALARY	ID_ORDER	DATE_ORDER	TYPE_ORDER	FEEDBACK	PHONE	DATE_PAY	PRICE
▶ 1	203487217	07/06/2000	moti	dan	7000	5	08/01/2020	sitting	3	504168272	08/01/2020	225
2	243432817	08/08/1996	tamar	somech	6600	3	02/04/2020	sitting	4	507496281	02/04/2020	52
3	433432117	02/08/1987	elroyi	cohen	4900	2	01/04/2020	sitting	5	508496284	01/04/2020	175
4	203432117	01/02/1997	shir	levi	9000	1	01/04/2020	sitting	5	506561604	01/04/2020	160
5	261792117	09/05/1990	lea	levi	6500	4	11/02/2020	sitting	4	529456118	11/02/2020	124
6	303432117	05/11/2002	miri	haron	5690	6	01/01/2020	take away	3	524165985	01/01/2020	260
7	203436948	08/10/1993	nir	levi	5500	7	10/02/2020	take away	2	508126483	10/02/2020	60
8	264987515	10/12/2005	moshe	cohen	9100	9	20/04/2020	take away	5	541652973	20/04/2020	225
9	216887584	10/05/2010	dor	dadon	8000	10	15/01/2020	take away	5	501659426	15/01/2020	80
10	203436948	08/10/1993	nir	levi	5500	11	10/02/2020	take away	2	508126483	10/02/2020	340
11	947232117	18/05/1998	shira	noach	6000	12	03/03/2020	take away	1	504592687	03/03/2020	60
12	216887584	10/05/2010	dor	dadon	8000	14	15/01/2020	take away	5	504592687	15/01/2020	160
13	216887584	10/05/2010	dor	dadon	8000	16	01/04/2020	sitting	5	508496284	01/04/2020	135
14	216887584	10/05/2010	dor	dadon	8000	17	01/07/2019	sitting	4	504592687	01/07/2019	155

שאלתה 1 על ה-VIEW

ההנהלה הציבה מסי' בדיקות לבדוק את רווחיות העסק, אחת מהן היא כשאר יש יותר מ-2 הזמנות ביום שמחירן מעל 100 שקלים.



The screenshot shows the SQL Developer interface with the following SQL query in the editor:

```
-- השאילתה בודקת לפי כל חודש את הימים בהם היו לפחות 2 הזמנות שמחירן מעל 100 ש"ח

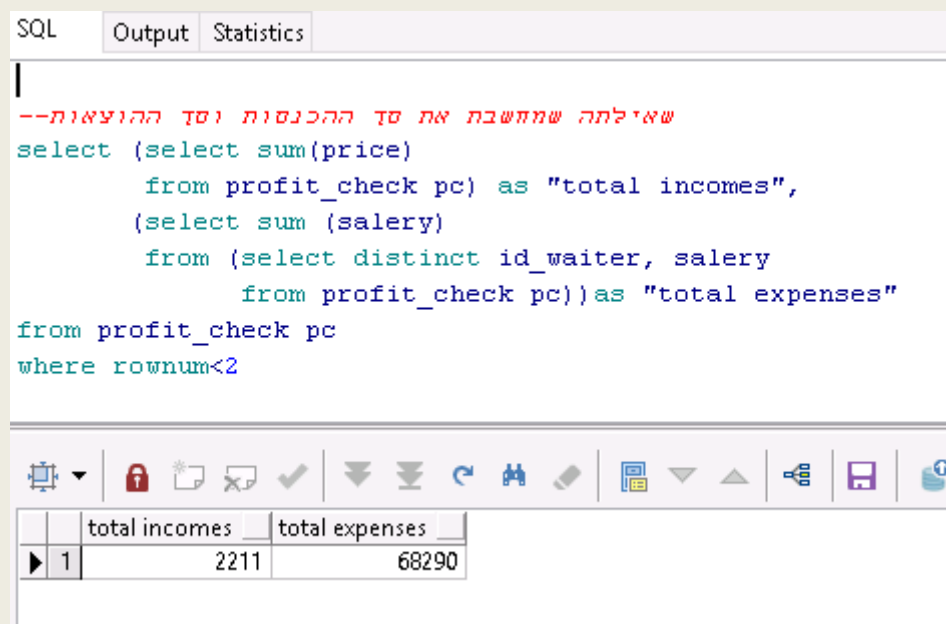
select date_order
from profit_check pc
where price>100
group by date_order
having count(id_order)>1
```

Below the editor, the 'Output' tab is active, displaying a table with the results of the query:

	DATE_ORDER
1	01/04/2020

שאלתה 2 על ה-VIEW

כדי שהעסק ייחשב רווחי בסוף כל חודש ההנהלה בודקת סך הוצאות אל מול סך ההכנסות ומחשבת את צעדיה בהתאם. השאילתה מחזירה את סך ההוצאות וסך ההכנסות של המסעדה.



The screenshot shows the SQL Developer interface with the following SQL query in the editor:

```
-- שאילתה שמחשבת את סך ההכנסות וסך ההוצאות

select (select sum(price)
        from profit_check pc) as "total incomes",
       (select sum(salary)
        from (select distinct id_waiter, salary
              from profit_check pc)) as "total expenses"
from profit_check pc
where rownum<2
```

Below the editor, the 'Output' tab is active, displaying a table with the results of the query:

	total incomes	total expenses
1	2211	68290

עדכון נתונים על ה-VIEW

בעקבות הפרש גדול בין ההכנסות להוצאות הנהלת המסעדה החליטה לעלות את סך מחיר ההזמנה ב-10% (ז"א המחיר לא כולל שרות)

SQL

Output

Statistics

-- עידכון על המבט

update profit_check

set price=price+(price*0.10)

	ID_WAITER	DOB	FNAME	LNAME	SALARY	ID_ORDER	DATE_ORDER	TYPE_ORDER	FEEDBACK	PHONE	DATE_PAY	PRICE
1	203487217	07/06/2000	moti	dan	7000	5	08/01/2020	sitting	3	504168272	08/01/2020	248
2	243432817	08/08/1996	tamar	somech	6600	3	02/04/2020	sitting	4	507496281	02/04/2020	57
3	433432117	02/08/1987	elroyi	cohen	4900	2	01/04/2020	sitting	5	508496284	01/04/2020	193
4	261792117	09/05/1990	lea	levi	6500	4	11/02/2020	sitting	4	529456118	11/02/2020	136
5	303432117	05/11/2002	miri	haron	5690	6	01/01/2020	take away	3	524165985	01/01/2020	286
6	203436948	08/10/1993	nir	levi	5500	7	10/02/2020	take away	2	508126483	10/02/2020	66
7	264987515	10/12/2005	moshe	cohen	9100	9	20/04/2020	take away	5	541652973	20/04/2020	248
8	216887584	10/05/2010	dor	dadon	8000	10	15/01/2020	take away	5	501659426	15/01/2020	88
9	203436948	08/10/1993	nir	levi	5500	11	10/02/2020	take away	2	508126483	10/02/2020	374
10	947232117	18/05/1998	shira	noach	6000	12	03/03/2020	take away	1	504592687	03/03/2020	66
11	216887584	10/05/2010	dor	dadon	8000	14	15/01/2020	take away	5	504592687	15/01/2020	176
12	216887584	10/05/2010	dor	dadon	8000	16	01/04/2020	sitting	5	508496284	01/04/2020	149
13	216887584	10/05/2010	dor	dadon	8000	17	01/07/2019	sitting	4	504592687	01/07/2019	171

ההשקפה לאחר העדכון. ניתן לראות שכל מחיר הזמנה עלה ב-10%.

מחיקת נתונים מה-VIEW

בעקבות הפרש גדול בין ההכנסות להוצאות הנהלת המסעדה החליטה לקצץ בשכר העובדים. מה שלא היה מקובל על עובד בעל ת.ז 203432117 ולכן ביקש להתפטר.

SQL

Output

Statistics

-- מחיקת נתונים מהמבט

delete from profit_check

where id_waiter='203432117' |

	ID_WAITER	DOB	FNAME	LNAME	SALARY	ID_ORDER	DATE_ORDER	TYPE_ORDER	FEEDBACK	PHONE	DATE_PAY	PRICE
1	203487217	07/06/2000	moti	dan	7000	5	08/01/2020	sitting	3	504168272	08/01/2020	248
2	243432817	08/08/1996	tamar	somech	6600	3	02/04/2020	sitting	4	507496281	02/04/2020	57
3	433432117	02/08/1987	elroyi	cohen	4900	2	01/04/2020	sitting	5	508496284	01/04/2020	193
4	261792117	09/05/1990	lea	levi	6500	4	11/02/2020	sitting	4	529456118	11/02/2020	136
5	303432117	05/11/2002	miri	haron	5690	6	01/01/2020	take away	3	524165985	01/01/2020	286
6	203436948	08/10/1993	nir	levi	5500	7	10/02/2020	take away	2	508126483	10/02/2020	66
7	264987515	10/12/2005	moshe	cohen	9100	9	20/04/2020	take away	5	541652973	20/04/2020	248
8	216887584	10/05/2010	dor	dadon	8000	10	15/01/2020	take away	5	501659426	15/01/2020	88
9	203436948	08/10/1993	nir	levi	5500	11	10/02/2020	take away	2	508126483	10/02/2020	374
10	947232117	18/05/1998	shira	noach	6000	12	03/03/2020	take away	1	504592687	03/03/2020	66
11	216887584	10/05/2010	dor	dadon	8000	14	15/01/2020	take away	5	504592687	15/01/2020	176
12	216887584	10/05/2010	dor	dadon	8000	16	01/04/2020	sitting	5	508496284	01/04/2020	149
13	216887584	10/05/2010	dor	dadon	8000	17	01/07/2019	sitting	4	504592687	01/07/2019	171

ההשקפה לאחר המחיקה. ניתן לראות שהמלצר בעל ת.ז הנ"ל נמחק מההשקפה.

משימה 5

5.1 שאלות עם פרמטרים

שאלתה1

השאלתה והתוצאה לאחר הכנסת הערכים שהתבקשנו

SQL Output Statistics

--1--
--השאלתה מחזירה עבור כל חודש את מס ההזמנות והסוג
--בטוח מחיר מסוים שיבוחר המשתמש ובסוג ההזמנה

select extract(month from date_order) as "Month",
count(id_order) as "Num of Orders",
type_order
from invite
where id_order in (select id_order
from payment
where price >(<name="enter a price"
type="integer"
hint="price should be only numbers">))
and type_order in (<name="choose type order"
list=" 'sitting',s , 'take away',t "
description="true"
required="true"
multiselect="true">)
group by extract(month from date_order),type_order
order by "Num of Orders" desc

	Month	Num of Orders	TYPE_ORDER
▶ 1	4	3	sitting
2	7	1	sitting
3	1	1	sitting
4	2	1	sitting

בחלון מצד ימין- כשהמשתמש בוחר ערכים הוא יכול לראות את ההערה שאומרת לו שעליו להכניס מחיר באמצעות מספרים.

בחלון האמצעי- ניתן לראות שהמשתמש חייב לבחור סוג הזמנה אחרת השאלתה לא תרוץ.

בחלון מצד שמאל- ניתן לראות את האפשרות לבחירת ערכים מרובים.

Select values

☒ s
☐ t

OK Cancel Clear

Variables

Name	Value
enter a price	100
choose type order	

OK Cancel Clear

choose type order is required

Variables

Name	Value
enter a price	100
choose type order	

OK Cancel Clear

price should be only numbers

שאלתה2


השאלתה והתוצאה לאחר הכנסת הערכים שהתבקשנו

SQL Output Statistics

--2--
--השאלתה מחזירה את פרטי הלקוח, חס ההזמנה שלו וכן את הסכום ששילם--
--אחרי שהמשתמש בוחר אם לראות את כל ההזמנות או רק הזמנות שהן מעל 100 שח--
--וכן ניתן למיין לפי שמות משפחה--

select invite.id_order , price ,customer.phone , lname, fname
from invite join customer
on invite.phone = customer.phone
join payment
on invite.id_order = payment.id_order
where price >(&<name="price">100?"
checkbox="100,0">)
and lname like(&<name="choose last name from the list"
type="string"
list="select lname from customer"
ifempty="%">)

order by price



	ID_ORDER	PRICE	PHONE	LNAME	FNAME
1	1	160	506561604	shalom	... michal ...
2	16	135	508496284	cohen	... yonatan ...
3	2	175	508496284	cohen	... yonatan ...
4	4	124	529456118	bloch	... miri ...
5	17	155	504592687	cocavi	... ela ...
6	14	160	504592687	cocavi	... ela ...
7	11	340	508126483	levi	... yonatan ...
8	6	260	524165985	cohen	... rona ...
9	5	225	504168272	amitay	... gali ...
10	9	225	541652973	shem-tov	... nir ...

ניתן לראות שלמרות שהמשתמש לא בחר ערך מתוך הרשימה עדיין חזרו רשומות
וזה בגלל הפרמטר ifempty שמחזיר את תוצאת השאלתה.

Variables

Name	Value
price>100?	<input checked="" type="checkbox"/>
choose last name from the list	

OK Cancel Clear

שאלתה3

השאלתה והתוצאה לאחר הכנסת הערכים שהתבקשנו

SQL Output Statistics

--3--
--השאלתה מחזירה משכורת של מלצר ואת התז שלו ע"י בחירת תאריך מסוים ושם המלצר מרשימה נתונה--

select salary , id_waiter
from waiter
where id_waiter in(select id_waiter
from invite
where date_order between to_date(&<name="from date"
type="date"
hint="date template dd/mm/yyyy">)
and to_date(&<name="to date"
type="date"
hint="date template dd/mm/yyyy">))
and lname in (&<name="waiter name"
type="string"
list="select lname from waiter"
multiselect="true"
required="true">)

	SALARY	ID_WAITER
1	4900	433432117
2	9100	264987515
3	8000	216887584

בחלון מצד ימין- בהכנסת הערכים ניתן לראות שהמשתמש מתבקש להכניס טווח תאריכים ויש לו "רמז" באיזו תבנית לרשום את התאריך.
בחלון האמצעי- המשתמש ניסה להריץ את השאלתה ללא הכנסת שם מלצר ולא הצליח. קופצת לו הערה שהמשתמש חייב לבחור לפחות שם אחד.
בחלון מצד שמאל- ניתן לראות את האופציה לבחירת ערכים מרובים של שמות

Select values

☐ levi
☐ cohen
☐ somech
☐ levi
☐ dan
☐ haron
☐ levi
☐ noach
☒ cohen
☒ dadon

OK Cancel Clear

Variables

Name	Value
from date	01/04/2020
to date	30/04/2020
waiter name	

OK Cancel Clear

Variables

Name	Value
from date	
to date	
waiter name	

OK Cancel Clear

waiter name is required

date template dd/mm/yyyy


שאילתה 4

השאילתה והתוצאה לאחר הכנסת הערכים שהתבקשנו

SQL Output Statistics

--4--
-- השאילתה מחזירה את מחיר הזמנה והתכולה שלה
-- לפי מס הזמנה שהמשתמש יבחר מתוך רשימת ההזמנות הנתונה
-- במידה ולא יבחר השאילתה תבחר הזמנה ברירת מחדל--

select payment.price ,
 containing.amount,
 containing.id_order,
 containing.id_dish
from payment join containing
on containing.id_order=payment.id_order
where containing.id_order in (<name="enter num of order"
 type="integer"
 list="select id_order from invite"
 default="select id_order from invite">)



	PRICE	AMOUNT	ID_ORDER	ID_DISH
▶ 1	160	2	1	10

Variables

Name	Value
enter num of order	1

OK Cancel Clear

5.2 הכנסת נתונים ע"י מחולל הנתונים DATA GENERATOR

טבלת waiter

WAITER				
Owner		Table		Number of records
SSHLOMI		WAITER		50
Name	Type	Size	Data	
ID_WAITER	VARCHAR2	10	[000000000]	
DOB	DATE		Random(01-01-1930, 01-01-2000)	
FNAME	VARCHAR2	20	FirstName	
LNAME	VARCHAR2	20	LastName	
SALARY	NUMBER	10	Random(4000, 10000)	
*				

SQL Output Statistics							
select * from waiter							
Select waiter Select customer Select dish Select waiter_phone Select invite Se							
	ID_WAITER	DOB	FNAME	LNAME	SALARY		
1	621517053	07/05/1981	Omar	Bell	7073		
2	854212141	11/09/1976	Daryl	Dunaway	4212		
3	301595214	19/12/1967	Carla	Loveless	7657		
4	165210410	26/01/1990	Alec	McCabe	9443		
5	427032235	13/04/1934	Dick	Saucedo	7637		
6	367764399	08/02/1998	Al	Lloyd	7097		
7	003291216	16/12/1973	Cuba	Cagle	5678		
8	225064702	04/01/1949	Gaby	Alda	4242		
9	671533319	10/12/1946	Lonnie	Linney	8509		
10	404431045	04/06/1984	Ashton	Sewell	7874		
11	460854915	09/02/1991	Kazem	Bragg	4140		
12	724133650	14/10/1956	Lea	Malkovich	9750		
13	811643836	04/07/1969	Lindsey	Carrington	9213		
14	411914436	31/12/1961	Jimmie	Grier	5422		
15	864986767	04/06/1934	Graham	McCabe	6189		
16	297626536	20/09/1947	Kevn	McConaughy	7280		
17	888459870	14/01/1983	Eddie	McPherson	5992		
18	358028188	17/01/1949	Elvis	Feuerstein	5341		
19	685802667	14/12/1968	Maxine	Dreyfuss	7070		
20	146974514	15/01/1968	Night	Chinlund	5263		
21	612988348	25/05/1934	Ty	Milsap	5700		
22	325786184	15/07/1982	Wes	O'Neill	6996		
23	612363503	30/03/1968	Frankie	Rodgers	6392		
24	168076039	08/11/1940	Jane	Leary	7007		
25	524316029	07/01/1990	Wesley	Beckinsale	6419		
26	936597742	28/02/1973	Jody	Wilkinson	8505		
27	782501546	30/08/1938	Treat	Craddock	6788		
28	603058212	26/03/1965	Beverley	Delta	6866		
29	285519207	30/11/1991	Mary	McLean	9998		
30	765669431	24/04/1947	Perry	Yankovic	5985		

customer טבלת

CUSTOMER

Owner

SSHLOMI

Table

CUSTOMER

Number of records

50

Name	Type	Size	Data
LNAME	VARCHAR2	20	LastName
FNAME	VARCHAR2	20	FirstName
PHONE	NUMBER	10	['05'00000000]
EMAIL	VARCHAR2	20	Email
CLUB_MEMBER	NUMBER	10	Sequence(Start, [Inc], [WithinParent])
ADDRESS	VARCHAR2	20	Address1
*			

SQL

Output

Statistics

select * from customer

טבלת dish

DISH				
Owner		Table		Number of records
SSHLOMI		DISH		50
Name	Type	Size	Data	
ID_DISH	VARCHAR2	10	Sequence(Start, [Inc], [WithinParent])	
PRICE	NUMBER	10	Random(10, 80)	
NAME	VARCHAR2	20	[aaaa]	
TYPE_DISH	VARCHAR2	20	List('first dish', 'main dish', 'last dish')	
*				

SQL

Output

Statistics


select * from dish

Select waiter

Select customer

Select dish

Select waiter_pl



		ID_DISH	PRICE	NAME	TYPE_DISH
▶	1	1	60	nsuj	last dish
	2	2	10	tgeh	last dish
	3	3	18	akqj	first dish
	4	4	36	hcnd	first dish
	5	5	70	lhqk	main dish
	6	6	18	yysq	main dish
	7	7	75	rduk	first dish
	8	8	75	twlh	main dish
	9	9	36	eiaw	main dish
	10	10	51	scnw	last dish
	11	11	70	qzhk	first dish
	12	12	64	pqew	last dish
	13	13	27	mgap	last dish
	14	14	40	iqrn	last dish
	15	15	71	pxls	last dish
	16	16	25	ympr	main dish
	17	17	51	jbwi	main dish
	18	18	73	svlt	last dish
	19	19	11	nkaa	main dish
	20	20	42	ufwx	last dish
	21	21	65	jgoh	last dish
	22	22	47	lbae	main dish
	23	23	27	rpyr	first dish
	24	24	31	mypx	main dish
	25	25	34	ancv	first dish
	26	26	79	zvig	main dish
	27	27	47	gaij	first dish
	28	28	55	bjoy	main dish
	29	29	65	ybmh	last dish
	30	30	60	ckqx	first dish
	31	31	27	icht	last dish

INVITE

Owner: SSHLOMI Table: INVITE Number of records: 50

Name	Type	Size	Data
ID_ORDER	VARCHAR2	10	Sequence(Start, [Inc], [WithinParent])
DATE_ORDER	DATE		Random(01-01-2010, 01-05-2020)
TYPE_ORDER	VARCHAR2	20	List('sitting', 'take away')
FEEDBACK	VARCHAR2	20	Random(1, 5)
ID_WAITER	VARCHAR2	10	List(select id_waiter from waiter)
PHONE	NUMBER	10	List(select phone from customer)
*			

SQL	Output	Statistics
<pre>select * from invite</pre>		
Select waiter	Select customer	Select dish
Select waiter_phone	Select invite	Select payment
ID_ORDER	DATE_ORDER	TYPE_ORDER
FEEDBACK	ID_WAITER	PHONE
1	04/12/2010	take away ... 3 ...
2	08/05/2016	sitting ... 4 ...
3	23/01/2015	take away ... 1 ...
4	14/03/2012	take away ... 3 ...
5	01/02/2017	take away ... 3 ...
6	02/10/2019	sitting ... 1 ...
7	18/09/2018	take away ... 4 ...
8	24/05/2011	sitting ... 1 ...
9	29/02/2016	take away ... 5 ...
10	15/08/2018	sitting ... 4 ...
11	26/12/2016	take away ... 5 ...
12	25/12/2016	take away ... 2 ...
13	05/02/2013	take away ... 4 ...
14	21/09/2014	take away ... 3 ...
15	15/01/2014	sitting ... 1 ...
16	10/02/2011	sitting ... 4 ...
17	11/03/2013	sitting ... 4 ...
18	18/03/2011	take away ... 3 ...
19	01/08/2017	sitting ... 5 ...
20	08/11/2014	sitting ... 1 ...
21	03/10/2013	sitting ... 3 ...
22	07/03/2010	sitting ... 3 ...
23	17/05/2017	sitting ... 2 ...
24	22/05/2018	take away ... 2 ...
25	20/07/2016	sitting ... 1 ...
26	04/07/2015	sitting ... 4 ...
27	02/03/2017	take away ... 5 ...
28	14/02/2016	take away ... 1 ...
29	24/10/2014	sitting ... 5 ...
30	11/02/2011	sitting ... 5 ...
31	13/09/2014	take away ... 1 ...

טבלת waiter_phone


WAITER_PHONE				
Owner	Table		Number of records	
SSHLOMI	WAITER_PHONE		50	
Name	Type	Size	Data	
PHONE	NUMBER	10	['05'00000000]	
ID_WAITER	VARCHAR2	10	List(select id_waiter from waiter)	
*				

SQL

Output

Statistics

from waiter_phone



	PHONE	ID_WAITER
1	585225025	411914436
2	506187130	879823873
3	581364407	936597742
4	577697821	621517053
5	560890202	325786184
6	511711414	297626536
7	581403090	696648340
8	555113160	811643836
9	558452133	685802667
10	528320545	524316029
11	549263487	805267755
12	584577107	696648340
13	555319674	060097597
14	587248866	404431045
15	576893498	059550329
16	563806148	783727486
17	530328889	524316029
18	546050976	621517053
19	563586686	573667375
20	510448927	366369296
21	516924493	225064702
22	565470498	612363503
23	543957044	783727486
24	537468816	765825585
25	544960451	864986767
26	530488820	325786184
27	542473324	573667375
28	517976894	059550329
29	552674232	367764399
30	549504564	146974514
31	556374504	805267755
32	542706224	358028188


טבלת payment

PAYMENT				
Owner	Table		Number of records	
SSHLOMI	PAYMENT		50	
Name	Type	Size	Data	
DATE_PAY	DATE		List(select date_order from invite)	
PRICE	NUMBER	10	Random(80, 1000)	
ID_ORDER	VARCHAR2	10	List(select id_order from invite)	

SQL				Output	Statistics
select * from payment					
		DATE_PAY	PRICE	ID_ORDER	
1	11/02/2011	83	12		
2	25/07/2014	740	40		
3	28/07/2010	933	49		
4	04/10/2012	231	36		
5	12/12/2011	516	35		
6	05/11/2019	270	20		
7	22/08/2019	119	30		
8	29/03/2013	976	46		
9	14/12/2014	847	44		
10	28/07/2010	309	12		
11	16/01/2016	572	24		
12	30/12/2015	613	41		
13	10/03/2017	515	18		
14	05/12/2012	558	30		
15	28/07/2010	718	39		
16	16/01/2016	221	19		
17	21/03/2017	341	20		
18	20/01/2017	108	28		
19	07/04/2015	289	41		
20	03/11/2012	423	47		
21	25/07/2014	229	30		
22	07/04/2015	769	27		
23	20/07/2018	448	14		
24	10/03/2017	162	36		
25	19/02/2018	827	30		
26	10/07/2015	839	32		
27	22/08/2019	475	2		
28	03/08/2012	817	12		
29	24/06/2012	285	8		
30	09/07/2012	443	24		
31	14/12/2014	369	12		

טבלת containing

CONTAINING				
◀	Owner	Table	Number of records	
▶	SSHLOMI	CONTAINING	50	
...				
	Name	Type	Size	Data
	AMOUNT	NUMBER	20	Random(0, 10)
	ID_ORDER	VARCHAR2	10	List(select id_order from invite)
▶	ID_DISH	VARCHAR2	10	List(select id_dish from dish)
*				

SQL				
Output				
Statistics				
select * from containing				
Select waiter				
Select customer				
Select dish				
Select wa				
				
		AMOUNT	ID_ORDER	ID_DISH
▶ 1		5	27	9
2		4	8	38
3		7	28	12
4		4	37	39
5		6	12	48
6		8	10	46
7		9	48	6
8		10	2	39
9		0	20	39
10		7	32	26
11		9	39	40
12		8	19	10
13		4	18	37
14		4	44	14
15		5	9	14
16		2	22	40
17		2	40	19
18		3	19	1
19		9	9	50
20		2	50	47
21		10	13	6
22		2	22	47
23		5	44	48
24		7	31	15
25		5	9	4
26		8	26	4
27		0	20	18
28		9	5	33
29		8	39	31
30		2	40	6
31		4	48	4

משימה 6

6.1 יבוא נתונים מתוך קובץ text importer

יבוא ראשון

יבוא נתונים מתוך קובץ שיצרתי בשם "waiter importer" עם סיומת text
לתוך טבלת waiter ע"י text importer

The screenshot shows the 'Text Import Wizard' in SQL Developer. The 'File Data' tab displays the raw data from 'waiter importer.txt'. The 'Configuration' section shows settings for importing 5 fields: id_waiter, DOB, fname, lname, and salary. The 'Field Start' and 'Field End' are configured with relative positions. The 'Result Preview' at the bottom shows the data being imported into the 'waiter' table.

id_waiter	DOB	fname	lname	salary
756483546	01-01-1990	moshe	levi	5000
645371909	09-08-2000	noy	shaul	5100
325365783	08-07-1984	eden	well	6400
203467285	01-06-1998	uri	chaim	6450
303451187	21-01-2000	tal	levi	8790
201127564	25-08-1989	omer	ball	8600
908546287	15-12-1988	shirel	yefet	9600

בחירת טבלת waiter

select.sql sql waiter importer.bt x

Data from Textfile Data to Oracle

General

Owner: Table:

Commit every...: ☒ Overwrite duplicates ☐ Delete records ☐ Truncate table ☐ Ignore duplicates

Initializing Script: ...

Finalizing Script: ...

Fields

Field1 id_waiter -> ID_WAITER
Field2 DOB -> DOB
Field3 fname -> FNAME
Field4 lname -> LNAME
Field5 salary -> SALARY

Field:
Fieldtype:
Create SQL
SQL function: ...
additional Oracle processing, for example: substr(%, 1, 20)

Result Preview

id_waiter	DOB	fname	lname	salary
756483546	01-01-1990	moshe	levi	5000
645371909	09-08-2000	noy	shaul	5100
325365783	08-07-1984	eden	well	6400
203467285	01-06-1998	uri	chaim	6450
303451187	21-01-2000	tal	levi	8790
201127564	25-08-1989	omer	ball	8600
908546287	15-12-1988	shirel	yefet	9600
657483928	11-11-1995	eli	falach	6700
546342517	05-03-1992	avi	shlom	7600
546382946	01-01-1980	or	gdot	8000

Import Import to Script Close sshlomi@labdbwin waiter importer.bt loaded, 1 KB Help

בחירת יבוא לתוך script והצגת ה-script

select.sql sql waiter importer.bt waiter importer.sql x

SQL Output Statistics

```

insert into waiter (id_waiter, dob, fname, lname, salary)
values ('756483546', to_date('01-01-1990', 'DD-MM-YYYY'), 'moshe', 'levi', 5000);

insert into waiter (id_waiter, dob, fname, lname, salary)
values ('645371909', to_date('09-08-2000', 'DD-MM-YYYY'), 'noy', 'shaul', 5100);

insert into waiter (id_waiter, dob, fname, lname, salary)
values ('325365783', to_date('08-07-1984', 'DD-MM-YYYY'), 'eden', 'well', 6400);

insert into waiter (id_waiter, dob, fname, lname, salary)
values ('203467285', to_date('01-06-1998', 'DD-MM-YYYY'), 'uri', 'chaim', 6450);

insert into waiter (id_waiter, dob, fname, lname, salary)
values ('303451187', to_date('21-01-2000', 'DD-MM-YYYY'), 'tal', 'levi', 8790);

insert into waiter (id_waiter, dob, fname, lname, salary)
values ('201127564', to_date('25-08-1989', 'DD-MM-YYYY'), 'omer', 'ball', 8600);

insert into waiter (id_waiter, dob, fname, lname, salary)
values ('908546287', to_date('15-12-1988', 'DD-MM-YYYY'), 'shirel', 'yefet', 9600);

insert into waiter (id_waiter, dob, fname, lname, salary)
values ('657483928', to_date('11-11-1995', 'DD-MM-YYYY'), 'eli', 'falach', 6700);

insert into waiter (id_waiter, dob, fname, lname, salary)
values ('546342517', to_date('05-03-1992', 'DD-MM-YYYY'), 'avi', 'shlom', 7600);

insert into waiter (id_waiter, dob, fname, lname, salary)
values ('546382946', to_date('01-01-1980', 'DD-MM-YYYY'), 'or', 'gdot', 8000);

commit;

```

טבלת waiter לאחר יבוא הנתונים מתוך הקובץ

SQL

Output

Statistics

```
select * from waiter
```

	ID_WAITER	DOB	FNAME	LNAME	SALARY
▶ 1	756483546	01/01/1990	moshe ...	levi ...	5000
2	645371909	09/08/2000	noy ...	shaul ...	5100
3	325365783	08/07/1984	eden ...	well ...	6400
4	203467285	01/06/1998	uri ...	chaim ...	6450
5	303451187	21/01/2000	tal ...	levi ...	8790
6	201127564	25/08/1989	omer ...	ball ...	8600
7	908546287	15/12/1988	shirel ...	yefet ...	9600
8	657483928	11/11/1995	eli ...	falach ...	6700
9	546342517	05/03/1992	avi ...	shlom ...	7600
10	546382946	01/01/1980	or ...	gdot ...	8000

יבוא שני

text יבוא נתונים מתוך קובץ שיצרתי בשם "dish importer" עם סיומת text importer טבלת ע"י

shiran.createetable.SQL

drop.sql.sql

select.sql.sql

dish importer.txt

Data from Textfile | Data to Oracle

File Data

Encoding

id_dish,price,name,type_dish
11,15,icecoffe,last dish
12,18,alfajores,last dish
13,20,bread,first dish
14,25,mini chips,first dish
15,30,pizza margarita,main dish
16,35,Cheese salad,main dish
17,40,Belgian waffles,last dish
18,45, potato gnocchi,main dish
19,50,Cheese gnocchi,main dish

Configuration

General

Fieldcount
4

☒ End at line-end
☒ Name in header
☒ Skip empty lines

Quote character
"
Comment line

Import lines
1 ..

Field1 (+0 .. ",") id_dish
Field2 (+0 .. ",") price
Field3 (+0 .. ",") name
Field4 (+0 .. ",") type_dish

Field Start

☐ Relative position
☐ Absolute position
☐ Character

Field End

☐ Length
☐ Character

Filter

Apply

Result Preview

id_dish	price	name	type_dish
11	15	icecoffe	last dish
12	18	alfajores	last dish
13	20	bread	first dish
14	25	mini chips	first dish
15	30	pizza margarita	main dish
16	35	Cheese salad	main dish
17	40	Belgian waffles	last dish

Import

Import to Script

Close

sshloimi@labdbwin → dish importer.txt loaded, 1 KB

Help

בחירת טבלת dish

shiran.createtable.SQL drop.sql.sql select.sql.sql dish importer.txt

Data from Textfile Data to Oracle

General

Owner: Table:

Commit every...: ☒ Overwrite duplicates ☐ Delete records ☐ Truncate table ☐ Ignore duplicates

Initializing Script: ...

Finalizing Script: ...

Fields

Field1 id_dish -> ID_DISH
Field2 price -> PRICE
Field3 name -> NAME
Field4 type_dish -> TYPE_DISH

Field:
Fieldtype:
Create SQL
SQL function: ...
additional Oracle processing, for example: substr(%, 1, 20)

Result Preview

id_dish	price	name	type_dish
11	15	icecoffe	last dish
12	18	alfajores	last dish
13	20	bread	first dish
14	25	mini chips	first dish
15	30	pizza margarita	main dish
16	35	Cheese salad	main dish
17	40	Belgian waffles	last dish

Import Import to Script Close sshlomi@labdbwin dish importer.txt loaded, 1 KB Help

הצגת הטבלה לאחר בחירת יבוא הקובץ היישר לתוך ה-database

SQL Output Statistics

select * from dish

	ID_DISH	PRICE	NAME	TYPE_DISH
1	11	15	icecoffe	last dish
2	12	18	alfajores	last dish
3	13	20	bread	first dish
4	14	25	mini chips	first dish
5	15	30	pizza margarita	main dish
6	16	35	Cheese salad	main dish
7	17	40	Belgian waffles	last dish
8	18	45	potato gnocchi	main dish
9	19	50	Cheese gnocchi	main dish
10	20	55	denis fish	main dish

יבוא נתונים מתוך קובץ שיצרתי בשם "containing importer" עם סיומת text
לתוך טבלת containing ע"י text importer

The screenshot shows the 'Import Wizard' window in SQL Developer. The 'Configuration' tab is active, showing settings for importing data from a text file into an Oracle database. The 'File Data' section shows the file 'containing importer.txt' with columns 'amount, id_order, id_dish'. The 'Configuration' section has the following settings:

- General:** Fieldcount: 3, End at line-end: checked, Name in header: checked, Skip empty lines: checked.
- Quote character:** "
- Comment line:** (empty)
- Import lines:** 1 .. (empty)
- Field Start:** Relative position (selected), Absolute position, Character.
- Field End:** Length (selected), Character.

The 'Result Preview' tab shows a table with 10 rows of data:

amount	id_order	id_dish
3	1	11
7	2	12
6	3	13
5	4	14
4	5	15
3	6	16
1	7	17
2	8	18
4	9	19
1	10	20

The bottom status bar shows '10 records imported in 0.047 seconds'.

בחירת טבלת containing

select.sql sql containing importer.txt x insert.sql sql

Data from Textfile Data to Oracle

General

Owner Table

Commit every... ☒ Overwrite duplicates ☐ Delete records ☐ Truncate table
☐ Ignore duplicates

Initializing Script ...

Finalizing Script ...

Fields

Field1 amount -> AMOUNT
 Field2 id_order -> ID_ORDER
 Field3 id_dish -> ID_DISH

Field
 Fieldtype

SQL function ...
 additional Oracle processing, for example: substr(#, 1, 20)

Result Preview

amount	id_order	id_dish
3	1	11
7	2	12
6	3	13
5	4	14
4	5	15
3	6	16
1	7	17
2	8	18
4	9	19
1	10	20

sshlomi@labdbwin 10 records imported in 0.047 seconds

הצגת הטבלה לאחר בחירת יבוא הקובץ היישר לתוך ה-database

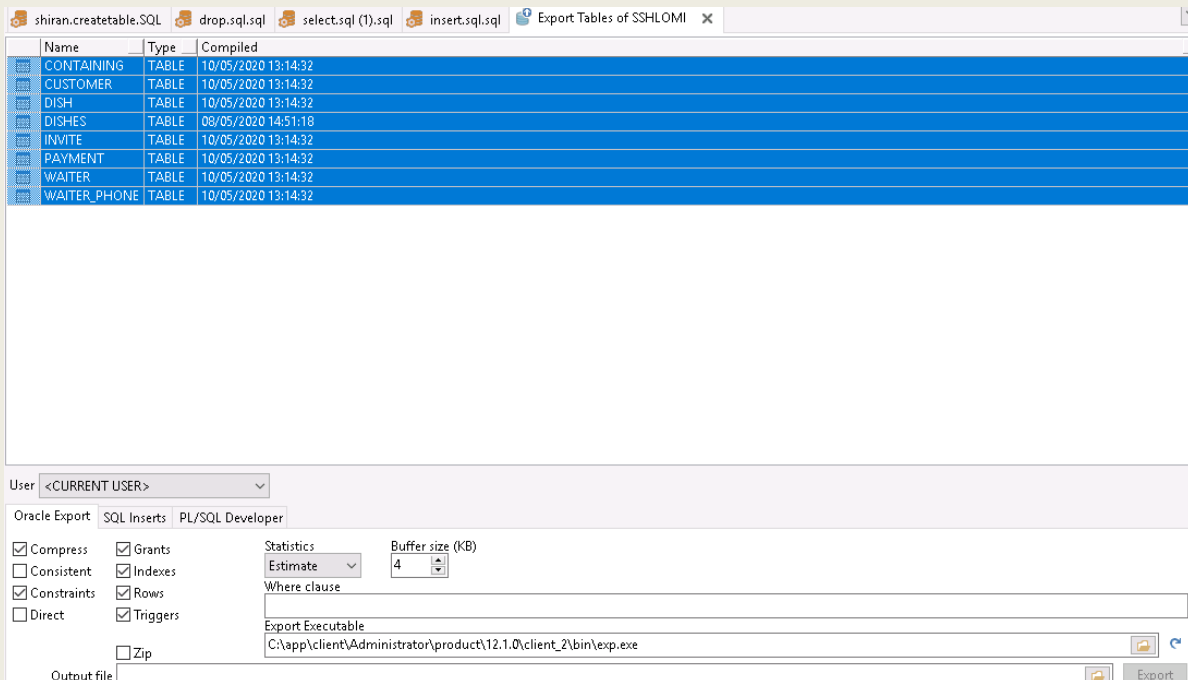
SQL Output Statistics

select * from containing

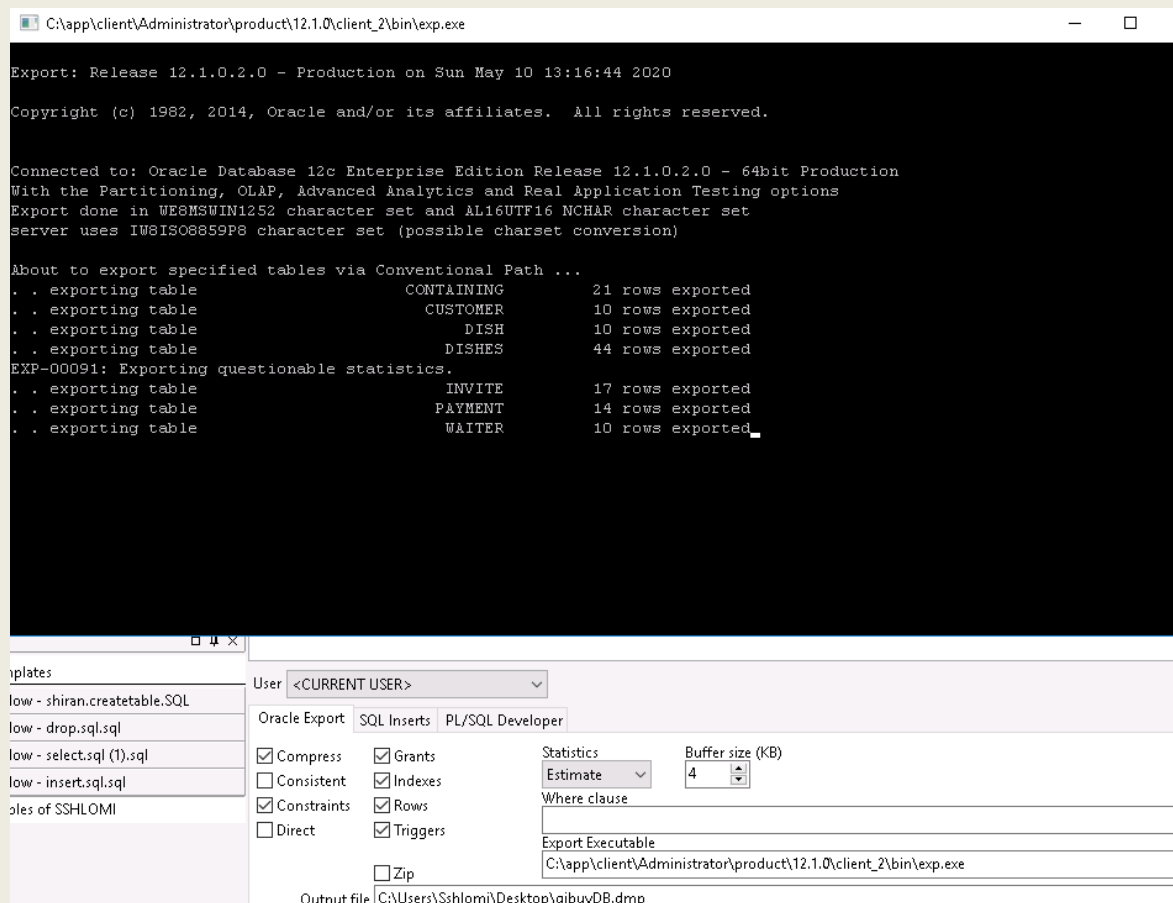
	AMOUNT	ID_ORDER	ID_DISH
1	3	1	11
2	7	2	12
3	6	3	13
4	5	4	14
5	4	5	15
6	3	6	16
7	1	7	17
8	2	8	18
9	4	9	19
10	1	10	20
11	1	2	2
12	3	2	1
13	1	3	3
14	2	4	5
15	5	5	4
16	4	10	6
17	2	7	8
18	1	8	9
19	3	9	10
20	2	6	2

6.2 גיבוי ושחזור נתונים

ביצוע גיבוי הנתונים ע"י export tables
ושמירת קובץ הגיבוי בשם gibuyDB



לחיצה על כפתור ה- export



ביצוע שחזור הנתונים ע"י import tables

shiran.createtable.SQL

drop.sql.sql

select.sql (1).sql

insert.sql.sql

Export Tables of SSHLOMI

Import Tables

Oracle Import

SQL Inserts

PL/SQL Developer

☒ Analyze

☐ Commit

☒ Constraints

☒ Grants

☒ Ignore

☒ Indexes

☒ Rows

☐ Show

Statistics

Always

Buffer size (KB)

30

From User

To User

Import Executable

C:\app\client\Administr

Import file

C:\Users\Sshlomi\Desktop\gibuyDB.dmp

בחירת קובץ הגיבוי ע"י import

C:\app\client\Administrator\product\12.1.0\client_2\bin\imp.exe

Import: Release 12.1.0.2.0 - Production on Sun May 10 13:18:12 2020
Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights reserved.
Connected to: Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options
Export file created by EXPORT:V12.01.00 via conventional path
import done in WE8MSWIN1252 character set and AL16UTF16 NCHAR character set
import server uses IWSISO8859P8 character set (possible charset conversion)
. importing SSHLOMI's objects into SSHLOMI
. importing SSHLOMI's objects into SSHLOMI
. . importing table "CONTAINING" 21 rows imported
. . importing table "CUSTOMER" 10 rows imported
. . importing table "DISH" 10 rows imported
. . importing table "DISHES" 44 rows imported
. . importing table "INVITE" 17 rows imported
. . importing table "PAYMENT" 14 rows imported
. . importing table "WAITER" 10 rows imported
. . importing table "WAITER_PHONE" 10 rows imported

shiran.createtable.SQL

drop.sql.sql

select.sql (1).sql

insert.sql.sql

if SSHLOMI

Import file

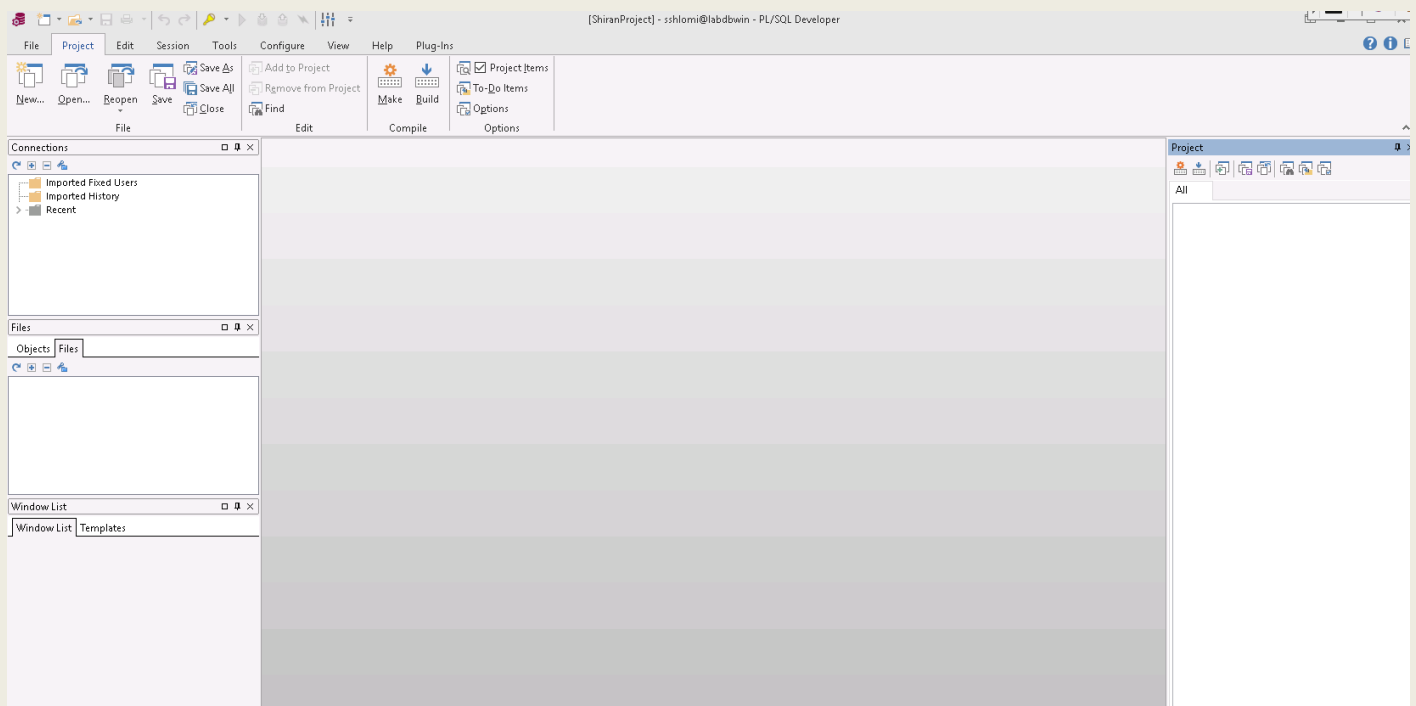
C:\Users\Sshlomi\Desktop\gibuyDB.dmp

שלב ב'

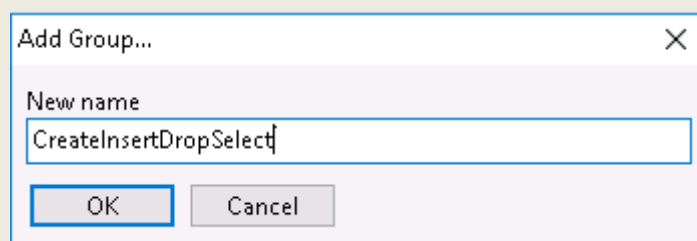
משימה 7

7.1 שימוש בפרויקט

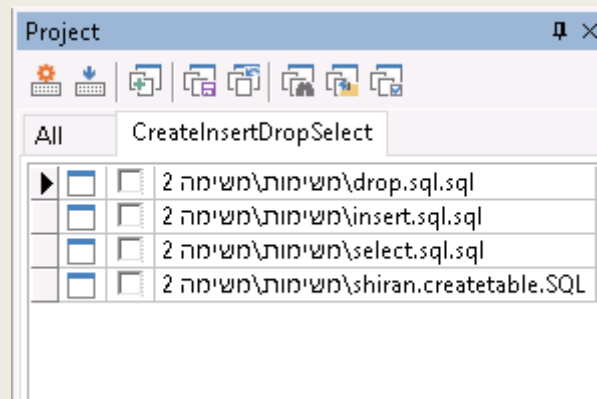
יצירת פרויקט חדש בלשונית project -> new
מצד ימין ניתן לראות את הפרויקט שנוצר (כרגע ריק ולא מקושר לאף קובץ)



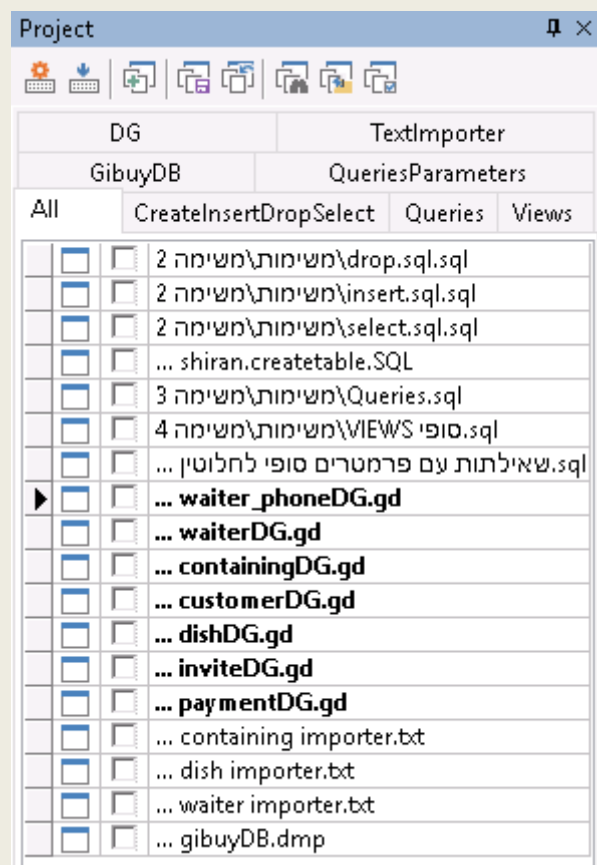
דוגמא להוספת קבוצה חדשה בשם CreateInsertDropSelect בפרויקט שיצרנו



יבוא קבצים לתוך הקבוצה החדשה שהוספנו לפרויקט מתוך תיקיית הקבצים שלי
(שעוסקים בפרויקט)



הצגה של פרויקט מלא לאחר קישור כל הקבצים מסודרים בקבוצות לפי משימות ושולבים



7.2 דוחות

יצירת הדוחות נעשו באמצעות קובץ מסוג report window

דוח 1

דוח המסכם עבור כל חודש את מס' ההזמנות שמחירן מעל 100 ₪

```
-- מספר ההזמנות עבור כל חודש שמחירן מעל 100

select extract(month from date_order) as "Month",
       count(id_order) as "Num of Orders"
from invite
where id_order in (select id_order
                   from payment
                   where price>100)
group by extract(month from date_order)
order by "Month"
```

הצגת הדוח לאחר ההרצה

Month	Num of Orders
1	5
2	5
3	2
4	13
5	2
6	2
7	3
8	1
9	3
11	1
12	1

הגדרת עיצוב הדוח בלשונית layout

-- מספר ההזמנות עבור כל חו ...								
Queries.sql shiran.createtable.SQL drop.sql.sql insert.sql.sql select.sql.sql								
SQL Layout Options								
Auto Update								
Item	Description	Style	Header	Align	Format	Break	Sum	
<input checked="" type="checkbox"/> Report Title		Blue Heade		Center				
<input checked="" type="checkbox"/> Variables		Default Fiel		Center				
Tabular Tables								
Form Tables								
Default Field								
<input checked="" type="checkbox"/> Month		Default Fiel	Blue Field	Center	0			
<input checked="" type="checkbox"/> Num of Orders			Blue Field	Center				

הצגת הדוח לאחר העיצוב

Month	Num of Orders
1	5
2	5
3	2
4	13
5	2
6	2
7	3
8	1
9	3
11	1
12	1

דוח 2

דוח פרוט כמות הזמנות עבור כל מלצר וסה"כ כמות ההזמנות שנלקחו

```

SQL Layout Options

-- דוח פרוט כמות הזמנות עבור כל מלצר

select lname,
       fname,
       waiter.id_waiter,
       count(id_order) as "total order",
       max(date_order) as "last order"

from waiter join invite
     on waiter.id_waiter = invite.id_waiter
group by waiter.lname , waiter.fname ,waiter.id_waiter

```

הצגת הדוח לאחר ההרצה



Lname	Fname	Id Waiter	total order	last order
Moody	Lena	952282643	2	21/02/2020
Delta	Joan	089521401	1	09/01/2016
levi	shir	203432117	2	05/04/2020
Buffalo	Olga	301081182	1	14/04/2012
Arden	Howie	363264549	2	09/03/2017
Gary	Chloe	911649009	1	03/04/2017
dan	moti	203487217	2	08/01/2020
Delta	Harold	466980984	1	30/04/2018
Skaggs	Melanie	515401143	3	10/09/2019
Rains	Chalee	709782951	2	01/11/2014
Williams	Kyra	915688082	1	16/01/2012
levi	nir	203436948	2	10/02/2020
Arjona	Terri	140226815	1	19/01/2014
noach	shira	947232117	4	03/03/2020
Lipnicki	Bobby	054181844	1	31/03/2016

הגדרות עיצוב הדוח בלשונית layout

SQL

Layout

Options

  ☐ Auto Update

Item	Description	Style	Header	Align	Format	Break	Sum
<input checked="" type="checkbox"/> Report Title	Waiter's Report	Blue Title		Center			
<input checked="" type="checkbox"/> Variables				None			
Tabular Tables				None			
Form Tables				None			
Default Field				None			
<input checked="" type="checkbox"/> LNAME		Default Fiel					
<input checked="" type="checkbox"/> FNAME							
<input checked="" type="checkbox"/> ID_WAITER							
<input checked="" type="checkbox"/> total order				Center			Sum per Report
<input checked="" type="checkbox"/> last order							

הצגת הדוח לאחר העיצוב (ניתן לראות בתחתית הדוח את סך ההזמנות)

Waiter's Report				
Lname	Fname	Id Waiter	total order	last order
Moody	Lena	952282643	2	21/02/2020
Delta	Joan	089521401	1	09/01/2016
levi	shir	203432117	2	05/04/2020
Buffalo	Olga	301081182	1	14/04/2012
Arden	Howie	363264549	2	09/03/2017
Gary	Chloe	911649009	1	03/04/2017
dan	moti	203487217	2	08/01/2020
Delta	Harold	466980984	1	30/04/2018
Skaggs	Melanie	515401143	3	10/09/2019
Rains	Chalee	709782951	2	01/11/2014
Williams	Kyra	915688082	1	16/01/2012
levi	nir	203436948	2	10/02/2020
Arjona	Terri	140226815	1	19/01/2014
noach	shira	947232117	4	03/03/2020
Lipnicki	Bobby	054181844	1	31/03/2016
Ruiz	Shannon	094144997	1	29/01/2011
cohen	elroyi	433432117	3	01/04/2020
levi	lea	261792117	1	11/02/2020
dadon	dor	216887584	5	01/04/2020
Hagerty	Pete	000201204	1	23/04/2016
somech	tamar	243432817	2	02/04/2020
cohen	moshe	264987515	6	20/04/2020
Reilly	Leo	246264185	2	30/04/2016
Bush	Claude	103903129	1	09/09/2015
Sharp	Bette	464444506	1	20/12/2013
Lindo	Kurt	061616596	2	19/04/2018
haron	miri	303432117	2	01/01/2020
Almond	Hal	816995641	2	01/04/2017
Brando	Kimberly	068815732	3	23/11/2019
Hong	Charlie	892216354	2	18/07/2015
Gellar	Dan	320772735	2	13/07/2018
Purefoy	Dylan	107678129	1	03/07/2017
Mirren	Delroy	961700479	2	13/06/2017
Plowright	Rosco	963246779	2	11/08/2012
			67	

דוח 3

דוח המציג את כל המנות של המסעדה על פי שם, סוג ומחיר

SQL	Layout	Options
select * from dish		
Id Dish	Price Name	Type Dish
1	45 pasta	main dish
2	40 pizza	main dish
3	52 fish and chips	main dish
4	45 salad	main dish
5	31 soup	first dish
6	20 cheese bread	first dish
7	27 cheesecake	last dish
8	30 apple pie	last dish
9	10 ice cream	last dish
10	25 chocolate cake	last dish

הגדרות עיצוב הדוח בלשונית layout

SQL

Layout

Options

☐ Auto Update

	Item	Description	Style	Header	Align	Format	Break	Sum
<input checked="" type="checkbox"/>	Report Title	M E N U	Blue Table		Center			
<input checked="" type="checkbox"/>	Variables							
	Tabular Tables							
	Form Tables							
	Default Field							
<input checked="" type="checkbox"/>	ID_DISH			Blue Heade	Center			
<input checked="" type="checkbox"/>	PRICE			Blue Heade	Center			
<input checked="" type="checkbox"/>	NAME			Blue Heade	Center			
<input checked="" type="checkbox"/>	TYPE DISH			Blue Heade	Center		Break + Header	

הצגת הדוח לאחר העיצוב :

MENU			
Id Dish	Price	Name	Type Dish
1	45	pasta	main dish
2	40	pizza	main dish
3	52	fish and chips	main dish
4	45	salad	main dish
Id Dish	Price	Name	Type Dish
5	31	soup	first dish
6	20	cheese bread	first dish
Id Dish	Price	Name	Type Dish
7	27	cheesecake	last dish
8	30	apple pie	last dish
9	10	ice cream	last dish
10	25	chocolate cake	last dish

7.3 גרפים

יצירת הגרפים נעשו באמצעות קובץ מסוג report window

גרף 1

גרף המייצג את מספר ההזמנות עבור כל חודש

```
SQL  Layout  Options

-- חס הזמנות עבור כל חודש

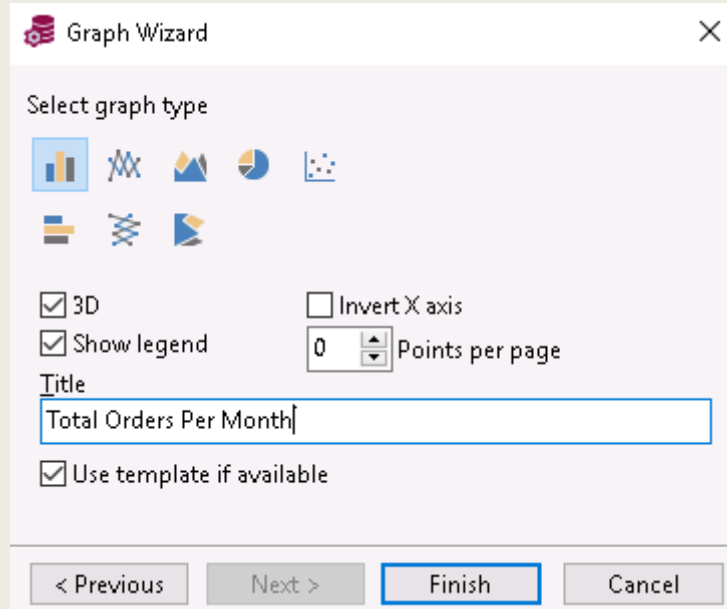
select extract(month from date_order) as "Month",
       count(id_order) as "Num of Orders"
from invite
group by extract(month from date_order)
order by "Month"
```

הצגת תוצאת השאילתה

Month	Num of Orders
1	11
2	6
3	5
4	16
5	3
6	2
7	8
8	4
9	6
11	4
12	2

כעת נבחר את סימבול הגרף (הסימבול האחרון מהרשימה) ובאמצעותו נגדיר איך יראה הגרף.

נבחר איזה עמודה תייצג את ציר ה- X ואיזה עמודה תייצג את ציר ה- Y .
נבחר את סוג הגרף, נבחר כותרת בהתאם וכן הצגה של המקרא (לשם נוחות).



Graph Wizard

Select graph type

☒ 3D ☐ Invert X axis

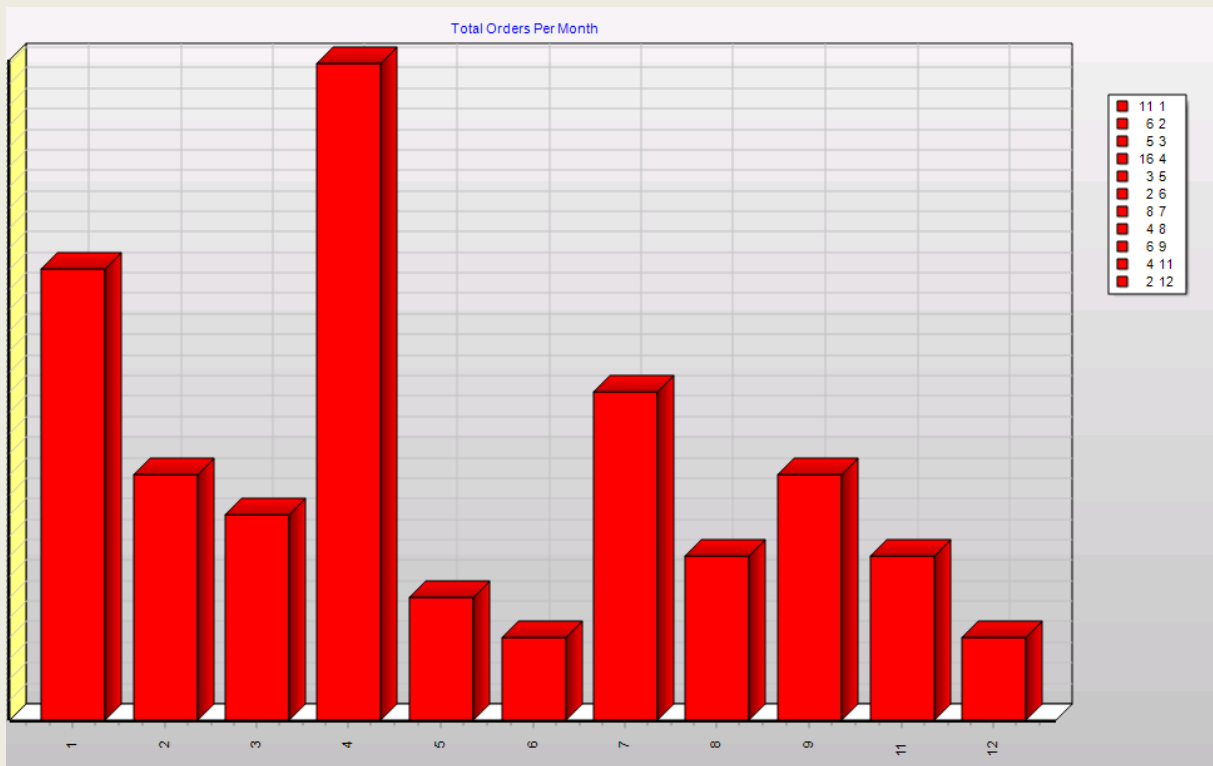
☒ Show legend Points per page

Title
Total Orders Per Month

☒ Use template if available

< Previous Next > Finish Cancel

הצגת הגרף:



גרף המייצג את כמות המנות שנמכרו עבור כל חודש

SQL

Layout

Options

```
-- כמות מנות עבור כל חודש


select extract(month from date_order) as "Month",
       sum(amount) as "Num of Dishes"
from invite i join containing c
on i.id_order=c.id_order
group by extract(month from date_order)
order by "Month"
```

הצגת תוצאת השאילתה






Month	Num of Dishes
1	48
2	30
3	27
4	74
5	1
6	24
7	24
8	22
9	13
11	13
12	8




כעת נבחר את סימבול הגרף (הסימבול האחרון מהרשימה) ובאמצעותו נגדיר איך יראה הגרף.

נבחר איזה עמודה תייצג את ציר ה- X ואיזה עמודה תייצג את ציר ה- Y .
נבחר את סוג הגרף, נבחר כותרת בהתאם וכן הצגה של המקרא (לשם נוחות).




Graph Wizard
✕

Select graph type

☒ 3D
 ☐ Invert X axis

☒ Show legend
 0   Points per page

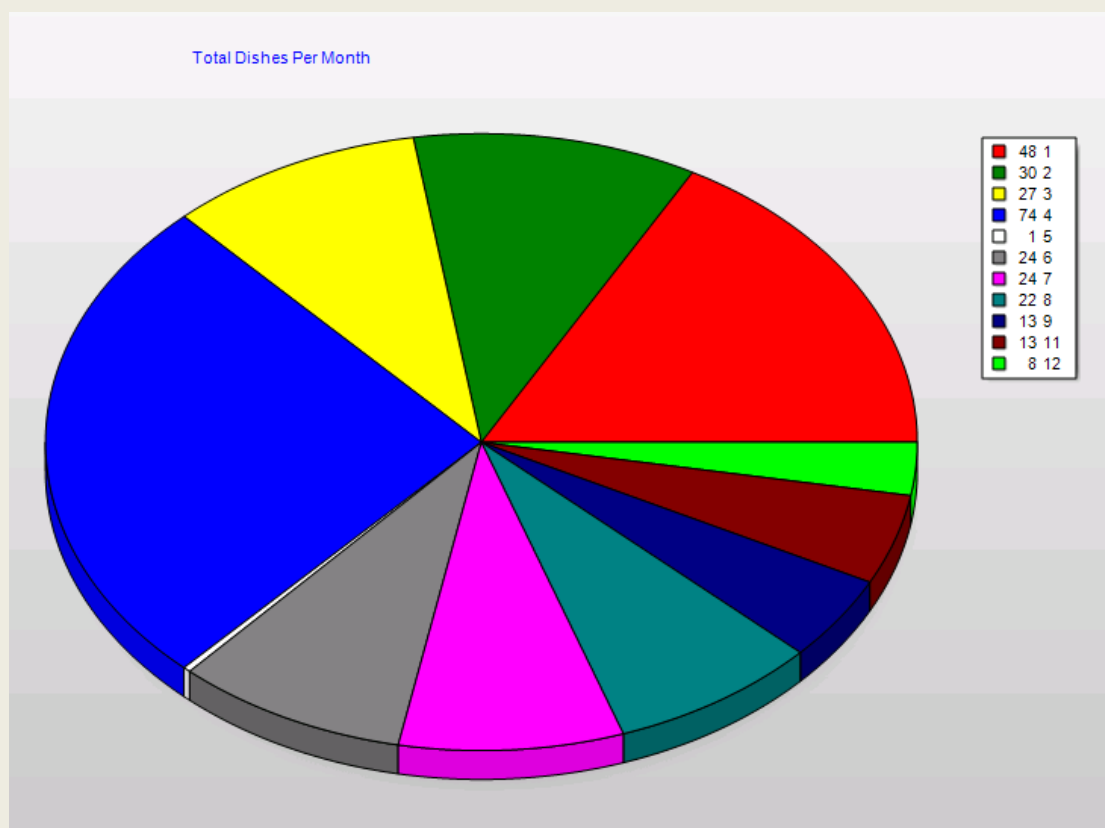
Title

Total Dishes Per Month

☒ Use template if available

< Previous
Next >
Finish
Cancel

הצגת הגרף :



משימה 8 – פקודת ALTER

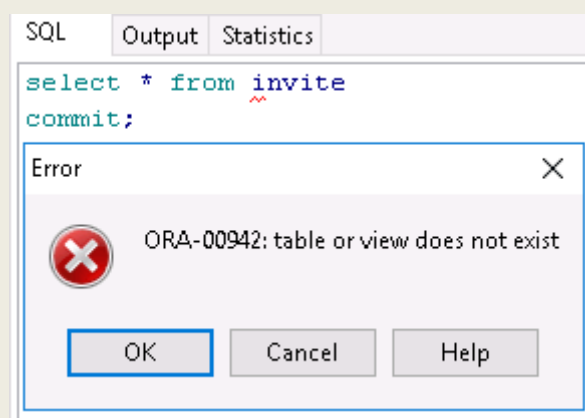
דוגמא 1:

שינוי השם של טבלת invite לשם orders

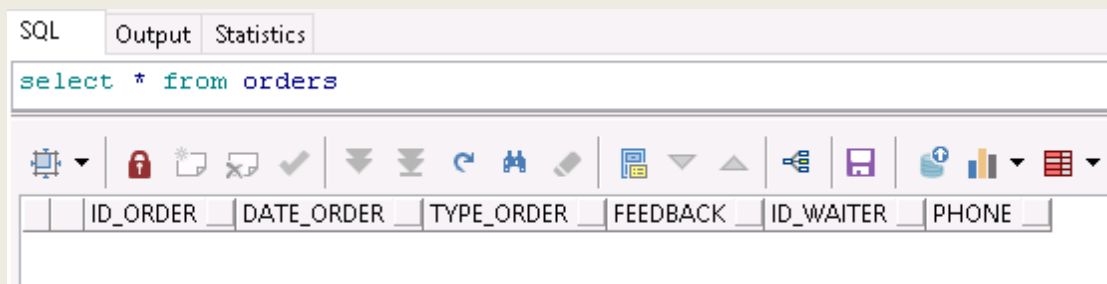
```
SQL Output Statistics
-- שם הנוי של טבלת הזמנות--

alter table invite
rename to orders
```

ניתן לראות ששליפת הטבלה בשמה הקודם לא צלחה



ואחרי שינוי השם של הטבלה פקודת select רצה בהצלחה.



דוגמא 2:

הוספת אילוץ לעמודה feedback

```
SQL Output Statistics
-- שם הנוי שדה קיים לאילוס--

alter table invite
add constraint feedback_check
check (feedback between 1 and 5)
```

ניסיון של הכנסת רשומה חדשה לטבלה לא הצליח מכיוון שהערך שהוכנס לשדה feedback הוא 9. והאילוץ הוא מס' בין 1-5.

SQL Output Statistics

```
insert into invite (id_order, date_order, type_order, feedback, id_waiter, phone)
values(170, to_date('01-07-2019','dd-mm-yyyy'), 'sitting', 9, 216887584, 0504592687);
```

Error

ORA-02290: check constraint (SSHLOMI.FEEDBACK_CHECK) violated

OK Cancel Help

דוגמא 3:

הוספת עמודה חדשה בשם num_memebers

SQL Output Statistics

```
-- הוספת עמודה לטבלת הזמנות
```

```
alter table invite
add num_members NUMBER(10)
```

הצגת הטבלה לאחר ההוספה. הפקודה צלחה למרות שלא הוכנסו הערכים כי הערך לא חייב להיות not null

SQL Output Statistics

```
select * from invite
```

	ID_ORDER	DATE_ORDER	TYPE_ORDER	FEEDBACK	ID_WAITER	PHONE	NUM_MEMBERS
1	1	01/04/2020	sitting	5	203432117	506561604	
2	2	01/04/2020	sitting	5	433432117	508496284	
3	3	02/04/2020	sitting	4	243432817	507496281	
4	4	11/02/2020	sitting	4	261792117	529456118	
5	5	08/01/2020	sitting	3	203487217	504168272	
6	6	01/01/2020	take away	3	303432117	524165985	
7	7	10/02/2020	take away	2	203436948	508126483	
8	8	03/03/2020	take away	1	947232117	504592687	
9	9	20/04/2020	take away	5	264987515	541652973	
10	10	15/01/2020	take away	5	216887584	501659426	
11	11	10/02/2020	take away	2	203436948	508126483	
12	12	03/03/2020	take away	1	947232117	504592687	
13	13	20/04/2020	take away	5	264987515	541652973	
14	14	15/01/2020	take away	5	216887584	504592687	


הוספת עמודה חדשה וכן אילוץ לאותה עמודה.

SQL	Output	Statistics
<pre>-- הוספת עמודה לטבלת תשלומים alter table payment add type_pay VARCHAR2(10) -- הוספת אילוץ לעמודה alter table payment add constraint type_pay_check check (type_pay in ('cash','credit'))</pre>		

ניסיון של הכנסת רשומה חדשה לא צלח מכיוון שהערך שהוכנס לשדה החדשה הינו gggg מה שלא תואם את האילוץ שאומר הערך צריך להיות cash או credit

SQL	Output	Statistics
<pre>insert into payment (date_pay, price, id_order,type_pay) values(to_date('08-01-2020','dd-mm-yyyy'),225 , 5 , 'gggg');</pre>		

Error



ORA-02290: check constraint (SSHLOMI.TYPE_PAY_CHECK) violated

OK Cancel Help

הצגת הטבלה לאחר הכנסת ערך מתאים לאילוף.


SQL

Output

Statistics

```
insert into payment (date_pay, price, id_order,type_pay)
values(to_date('08-01-2020','dd-mm-yyyy'),225 , 5 , 'cash');

select * from payment
```



		DATE_PAY	PRICE	ID_ORDER	TYPE_PAY
▶	1	08/01/2020	225	5	
	2	02/04/2020	52	3	
	3	01/04/2020	175	2	
	4	01/04/2020	160	1	
	5	11/02/2020	124	4	
	6	01/01/2020	260	6	
	7	10/02/2020	60	7	
	8	20/04/2020	225	9	
	9	15/01/2020	80	10	
	10	10/02/2020	340	11	
	11	03/03/2020	60	12	
	12	15/01/2020	160	14	
	13	01/04/2020	135	16	
	14	01/07/2019	155	17	
	15	08/01/2020	225	5	cash

דוגמא 5:

הוספת עמודה שאינה null

SQL		Output	Statistics
		<pre>-- הוספת עמודה לטבלת לקוחות-- alter table customer add city VARCHAR2(10) NOT NULL</pre>	

ניסון הכנסת נתונים לטבלה לא מצליח מכיוון שלא הכנסנו ערכים לעמודה החדשה והשגיאה אומרת שאי אפשר שהערך של העמודה city יהיה ריק.

SQL		Output	Statistics
		<pre>insert into customer (lname, fname, phone, email, club_member, address) values('shalom','michal',0506561604,'mic@gmail.com',898,'89 derech amelech'); insert values(insert values(insert values(insert values(insert values(</pre>	
		<pre>address) nachem begin'); address) nethaniho'); address) ach'); address) guryon');</pre>	

התוצאה לאחר הכנסת רשומה חדשה עם ערך מתאים לעמודה החדשה city

```
SQL      Output  Statistics

insert into customer (lname, fname, phone, email, club_member, address, city)
values ('shalom', 'michal', 0506561604, 'mic@gmail.com', 898, '89 derech amelech', 'givat shmuel');
select * from customer
commit;
```

Navigation icons: back, forward, search, etc.

	LNAM	FNAM	PHONE	EMAIL	CLUB_MEMBER	ADDRESS	CITY
1	shalom	michal	506561604	mic@gmail.com	898	89 derech amelech	givat shmuel

9.1 – הרשאת גישה ע"י פקודת GRANT וביטול גישה ע"י פקודת REVOKE

הרשאת גישה למשתמש newuser עבור:

הכנסת נתונים לטבלת waiter

SQL	Output	Statistics
<code>grant insert on waiter to newuser</code>		

מחיקת נתונים מטבלת dish

SQL	Output	Statistics
<code>grant delete on dish to newuser</code>		

כל הפעולות האפשריות מטבלת customer

SQL	Output	Statistics
<code>grant all on customer to newuser</code>		

הרשאת גישה ציבורית עבור שליפת נתונים מטבלת dish

SQL	Output	Statistics
<code>grant select on dish to public</code>		

ביטול הרשאת גישה למשתמש newuser עבור:

הכנסת נתונים לטבלת waiter

SQL	Output	Statistics
<code>revoke insert on waiter from newuser</code>		

מחיקת נתונים מטבלת dish

SQL	Output	Statistics
<code>revoke delete on dish from newuser</code>		

כל ההרשאות מטבלת customer

SQL	Output	Statistics
<code>revoke all on customer from newuser</code>		

ביטול הרשאת גישה ציבורית עבור שליפת נתונים מטבלת dish

SQL	Output	Statistics
<code>revoke select on dish from public</code>		

9.2 – פקודות COMMIT ו-ROLLBACK

COMMIT – פקודה זו מכניסה באופן סופי את השינויים למערכת. פעולה זו מתבצעת בסיום של כל טרנזקציה ולאחריה לא ניתן לחזור למה שהיה קודם לכן גם אם נכתוב את פקודת rollback.

ROLLBACK – הפקודה מאפשרת לנו לחזור אחורה לאחר ביצוע שינויים בבסיס הנתונים.

דוגמא 1:

ניתן לראות את תוצאת השאילתה לאחר הרצתה.

SQL

Output

Statistics

--הצגת פרטי לקוח ששילם מעל 100 שקלים עבור הזמנה--
--השאילתה מחזירה טלפון ושם לקוח שנעשה הזמנה שסכומה עולה על 100 שקלים וכן את הסכום ששילם--

select invite.id_order , price ,customer.phone , lname, fname
from invite join customer
on invite.phone = customer.phone
join payment
on invite.id_order = payment.id_order
where (price>100)
order by price

	ID_ORDER	PRICE	PHONE	LNAME	FNAME
1	4	124	529456118	bloch	miri
2	16	135	508496284	cohen	yonatan
3	17	155	504592687	cocavi	ela
4	1	160	506561604	shalom	michal
5	14	160	504592687	cocavi	ela
6	2	175	508496284	cohen	yonatan
7	9	225	541652973	shem-tov	nir
8	5	225	504168272	amitay	gali
9	6	260	524165985	cohen	rona
10	11	340	508126483	levi	yonatan

כעת נבצע עדכון של הוספת 10 שקלים כשרות למחיר הסופי של כל הזמנה וכך תיראה השאילתה לאחר העדכון:

SQL

Output

Statistics

update payment

set price=price+10

	ID_ORDER	PRICE	PHONE	LNAME	FNAME
1	4	134	529456118	bloch	miri
2	16	145	508496284	cohen	yonatan
3	17	165	504592687	cocavi	ela
4	1	170	506561604	shalom	michal
5	14	170	504592687	cocavi	ela
6	2	185	508496284	cohen	yonatan
7	9	235	541652973	shem-tov	nir
8	5	235	504168272	amitay	gali
9	6	270	524165985	cohen	rona
10	11	350	508126483	levi	yonatan


כעת נריץ את פקודת rollback ושוב נריץ את השאילתה ויהיה ניתן לראות שהצלחנו לחזור לאחור והעדכון לא נשמר.

SQL

Output

Statistics

rollback



		ID_ORDER	PRICE	PHONE	LNAME	FNAME
▶	1	4	124	529456118	bloch	miri
	2	16	135	508496284	cohen	yonatan
	3	17	155	504592687	cocavi	ela
	4	1	160	506561604	shalom	micah
	5	14	160	504592687	cocavi	ela
	6	2	175	508496284	cohen	yonatan
	7	9	225	541652973	shem-tov	nir
	8	5	225	504168272	amitay	gali
	9	6	260	524165985	cohen	rona
	10	11	340	508126483	levi	yonatan

נריץ שוב את שאילתת העדכון , ולאחריה את פקודת commit ולאחר מכן את פקודת rollback - כפי שניתן לראות העדכון נשמר והמחיר עלה ב-10 ₪.


SQL

Output

Statistics

commit

rollback



		ID_ORDER	PRICE	PHONE	LNAME	FNAME
▶	1	4	134	529456118	bloch	miri
	2	16	145	508496284	cohen	yonatan
	3	17	165	504592687	cocavi	ela
	4	1	170	506561604	shalom	micah
	5	14	170	504592687	cocavi	ela
	6	2	185	508496284	cohen	yonatan
	7	9	235	541652973	shem-tov	nir
	8	5	235	504168272	amitay	gali
	9	6	270	524165985	cohen	rona
	10	11	350	508126483	levi	yonatan

דוגמא 2 :

תוצאת השאילתה לאחר הרצתה.

SQL Output Statistics

-- העגת סכום המשכורות לחודש אפריל --

select sum(salary) as "April salary"
from waiter
where id_waiter in (select id_waiter from invite
 where date_order between to_date('01-04-2020','dd-mm-yyyy')
 and to_date('30-04-2020','dd-mm-yyyy'))

	April salary
1	37600

מחיקת מלצר (שסיים את עבודתו במסעדה) מתוך טבלת מלצרים .
ולאחר מכן שוב הרצה של השאילתה וניתן לראות שסכום המשכורות השתנה.

SQL Output Statistics

-- מחיקת רשומה --

delete from waiter
where id_waiter='203432117'

select sum(salary) as "April salary"
from waiter
where id_waiter in (select id_waiter from invite
 where date_order between to_date('01-04-2020','dd-mm-yyyy')
 and to_date('30-04-2020','dd-mm-yyyy'))

	April salary
1	28600

כעת נבצע פקודת rollback ושוב נריץ את השאילתה, יהיה ניתן לראות שהצלחנו
לחזור לאחור לפני השינוי של הנתונים . המחיקה לא נשמרה.

SQL Output Statistics

rollback

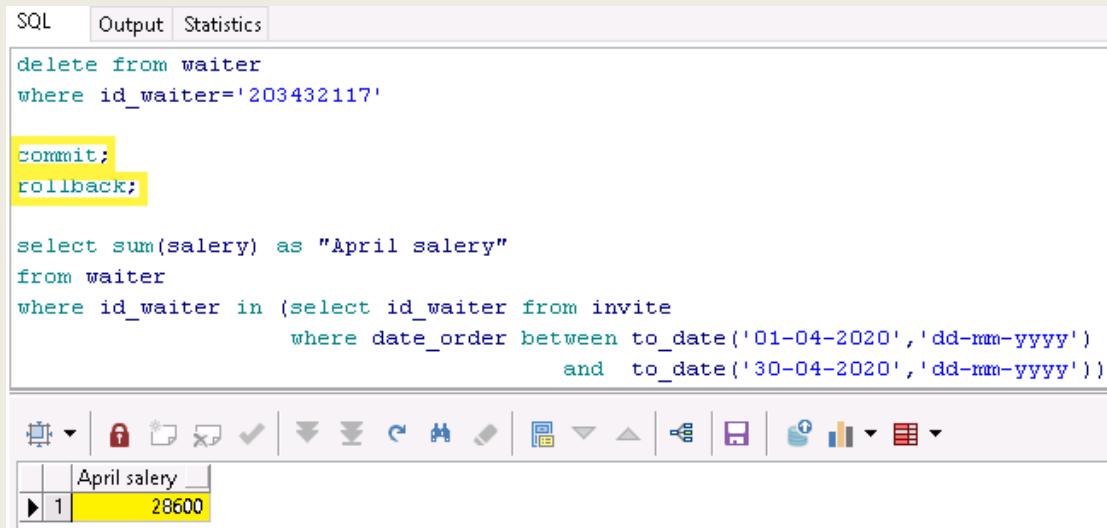
select sum(salary) as "April salary"
from waiter
where id_waiter in (select id_waiter from invite
 where date_order between to_date('01-04-2020','dd-mm-yyyy')
 and to_date('30-04-2020','dd-mm-yyyy'))

	April salary
1	37600

שוב נבצע את השאילתה למחיקת רשומה ולאחריה נריץ את פקודת commit כדי שישמרו השינויים בבסיס הנתונים .

לאחר מכן נריץ את פקודת rollback ושוב את השאילתה.

ניתן לראות שלמרות הרצת הפקודה rollback השינויים שביצענו קודם לכן לא נשמרו וזה מכיוון שקודם לכן דאגנו לרשום את פקודת commit .



```
delete from waiter
where id_waiter='203432117'

commit;
rollback;

select sum(salary) as "April salary"
from waiter
where id_waiter in (select id_waiter from invite
                    where date_order between to_date('01-04-2020','dd-mm-yyyy')
                    and to_date('30-04-2020','dd-mm-yyyy'))
```

	April salary
1	28600

דוגמא 3 :

כפי שניתן לראות בקוד – 2 פקודות של הכנסת נתונים לתוך טבלה ולאחר מכן פקודה האומרת "נקודת שמירה" בשם save1, ולאחריה עוד פקודות הכנסה של נתונים לתוך אותה טבלה. כשאר נשלף את הנתונים של אותה טבלה ניתן לראות את 4 הרשומות שהכנסנו.


SQL Output Statistics

```
insert into payment (date_pay, price, id_order)
values(to_date('03-03-2020','dd-mm-yyyy'),60 , 12 );
insert into payment (date_pay, price, id_order)
values(to_date('15-01-2020','dd-mm-yyyy'),160 , 14);

savepoint save1

insert into payment (date_pay, price, id_order)
values(to_date('01-04-2020','dd-mm-yyyy'),135 , 16 );
insert into payment (date_pay, price, id_order)
values(to_date('01-07-2019','dd-mm-yyyy'),155 , 17 );

select * from payment
```




		DATE_PAY	PRICE	ID_ORDER
▶	1	03/03/2020	60	12
	2	15/01/2020	160	14
	3	01/04/2020	135	16
	4	01/07/2019	155	17

כעת נריץ את פקודת rollback to save1 - ז"א תחזור אחורה לאותה נקודת שמירה שהגדרנו . וכפי שניתן לראות הנתונים נשמרו עד אותה נקודת שמירה ופקודת rollback לא חזרה אחורה אוטו ל- commit האחרון.

SQL Output Statistics

```
rollback to save1

select * from payment
```



		DATE_PAY	PRICE	ID_ORDER
▶	1	03/03/2020	60	12
	2	15/01/2020	160	14

משימה 10 – פרוצדורות

הרעיון הכללי:

בעל המסעדה החליט לנקוט במספר צעדים כדי לשמר את הלקוחות/להגדיל את מס' הלקוחות והרווח של המסעדה.

פרוצדורה 1

התוכנית בודקת את כמות ההזמנות של אותו לקוח שהמשתמש הכניס ומקנה לו הנחה על ההזמנה בהתאם.

האילוצים שהתוכנית בודקת :

- (1) הלקוח קיים במערכת
- (2) סך ההזמנות של הלקוח גדולה מ-1
- (3) מחיר כל הזמנה גדול מ-150 שקלים

במידה והלקוח מקיים את התנאים הוא יקבל הודעה עבור ההזמנות שלו והמחיר לאחר ההנחה.

אחרת, תוצג הודעה בהתאם.

```

create or replace procedure GetYourDiscount (CustomerPhone in
number,Thedate out date)
is

    strLN varchar2(15) NOT NULL := 'ln';
    strFN varchar2(15) NOT NULL := 'fn';

    countOrders number;

    thePrice payment.price$type;
    NewPrice number;

    cursor allId_order is
    select *
    from invite;

    --record type of exist table
    id_orderRec allId_order$rowtype;

    TheOrder varchar2 (15);

begin

    --for evrey record in invite table
    for id_orderRec in allId_order
    loop
        if(id_orderRec.Phone=CustomerPhone)
        then
            --select the name of customer that the user entered
            as a parameter
            select lname,fname into strLN,strFN
            from customer
            where phone=CustomerPhone;
            end if;
        end loop;
        --if the customer exists do-
        if(SQL%found)
        then
            --print the detailes of the customer
            Thedate:=sysdate;
            dbms_output.put_line('The date is: '|| Thedate);
            dbms_output.put_line('The name of customer is: ' ||
                                strLN || ' ' || strFN );
            dbms_output.put_line('The phone number is: ' ||
                                CustomerPhone );
            dbms_output.put_line('*****');

            --count the num of orders for that customer
            select count(id_order)into countOrders
            from invite
            where phone=CustomerPhone;

            --if there is at least 1 order do-
            if (countOrders>1)
            then
                --print some messages to the user
                dbms_output.put_line('Your amount of orders is'
                                || ' ' || countOrders );
                dbms_output.put_line('Maybe you will deserve
                                a discount...');
                dbms_output.put_line('*****');
            end if;
        end if;
    end loop;
end;

```



```

dbms_output.put_line('After testing:');
dbms_output.new_line;
--for every record it takes the id_order and puts
  in new var
for id_orderRec in allId_order
loop
  TheOrder:=id_orderRec.Id_Order;
  --select the price of the order
  select price into thePrice
  from payment
  where id_order=TheOrder;
  --if the price>150 do-
  if thePrice>150
  then
    --print the price befor the discount
    dbms_output.put_line('You deserve a discount
                        for order '||TheOrder
                        ||' '|| 'that costs
                        is'|| ' '||thePrice||' '|| 'shekels');

    --update the price for this order
    update payment
    set price=price-(price*0.2)
    where id_order=TheOrder;
    --commit;
    --select the price after the update/discount
    select price into NewPrice
    from payment
    where id_order=TheOrder;

    --print the price after the discount
    dbms_output.put_line('Your price after the
discount is' ||' '||NewPrice||' '|| 'shekels');

    --if the price<150
  else
    --print
    --dbms_output.put_line('*-----*');
    dbms_output.put_line('We are sorry. you do not
                        deserve a discount for order '||TheOrder);
  end if;
  dbms_output.put_line('*-----*');
end loop;
else
  --count of the orders<1
  dbms_output.put_line('We are sorry. you do not deserve
                        a discount for your order');
  end if;
--if the customer does not exists do-
else
  dbms_output.put_line('We are sorry.The customer does not exist');
end if;
end GetYourDiscount;|

```

הרצות:

דוגמא 1

Test script DBMS Output Statistics Profiler Trace

```

1 begin
2   -- Call the procedure
3   getyourdiscount(customerphone => :customerphone,
4                     thedate => :thedata);
5 end;

```

	Variable	Type	Value
▶	customerphone	Float	504592687
▶	thedata	Date	11/06/2020 13:38:58
*			

Test script DBMS Output Statistics Profiler Trace

Clear Buffer size 10000 ☒ Enabled

```

The date is:11-JUN-20
The name of customer is:cocavi ela
The phone number is:504592687
*****
Your amount of orders is 5
Maybe you will deserve a discount...
*****
After testing:

We are sorry. you do not deserve a discount for order 8
*-----*
We are sorry. you do not deserve a discount for order 12
*-----*
You deserve a discount for order 14 that costs is 160 shekels
Your price after the discount is 128 shekels
*-----*
You deserve a discount for order 15 that costs is 195 shekels
Your price after the discount is 156 shekels
*-----*
You deserve a discount for order 17 that costs is 155 shekels
Your price after the discount is 124 shekels
*-----*

```

ללקוח ישנן 5 הזמנות וניתן לראות בתמונה מצד שמאל את השיניים שנעשו לאחר ההנחה עבור ההזמנות שעמדו באילוף.

אחרי-

	ID_ORDER	PRICE
▶ 1	12	60
2	14	128
3	17	124
4	8	137
5	15	156

לפני-

	ID_ORDER	PRICE
▶ 1	12	60
2	14	160
3	17	155
4	8	137
5	15	195

דוגמא 2

The screenshot shows the SQL Developer interface with the 'Test script' tab selected. The script contains a PL/SQL block that calls a procedure named 'getyourdiscount' with parameters 'customerphone' and 'thedata'. The output table shows the values for these variables: 'customerphone' is 529456118 and 'thedata' is 11/06/2020 13:53:45. Below the table, the output text shows the date, customer name, and phone number, followed by a message: 'We are sorry. you do not deserve a discount for your order'.

```
Test script DBMS Output Statistics Profiler Trace
```

```
1 begin
2   -- Call the procedure
3   getyourdiscount(customerphone => :customerphone,
4                   thedate => :thedata);
5 end;
```

	Variable	Type	Value
▶	customerphone	Float	529456118
▶	thedata	Date	11/06/2020 13:53:45

Test script DBMS Output Statistics Profiler Trace

Clear Buffer size 10000 ☒ Enabled

The date is:11-JUN-20
The name of customer is:bloch miri
The phone number is:529456118

We are sorry. you do not deserve a discount for your order

ללקוח ישנה הזמנה אחת, הוא לא מקיים את האילוץ הראשון (מה שאומר שהתוכנית לא תיכנס לבדוק את האילוץ השני), ולכן הוא לא קיבל הנחה על ההזמנה שלו.

דוגמא 3

The screenshot shows the SQL Developer interface with the 'Test script' tab selected. The script is identical to the one in the previous example. The output table shows the values for the variables: 'customerphone' is 505464654 and 'thedata' is 11/06/2020 13:53:45. Below the table, the output text shows the date, customer name, and phone number, followed by a message: 'We are sorry.The customer does not exist'.

```
Test script DBMS Output Statistics Profiler Trace
```

```
1 begin
2   -- Call the procedure
3   getyourdiscount(customerphone => :customerphone,
4                   thedate => :thedata);
5 end;
```

	Variable	Type	Value
▶	customerphone	Float	505464654
▶	thedata	Date	11/06/2020 13:53:45

Test script DBMS Output Statistics Profiler Trace

Clear Buffer size 10000 ☒ Enabled

We are sorry.The customer does not exist

המשתמש הכניס מספר לקוח שלא קיים.

פרוצדורה 2

התוכנית מגרילה בין ההזמנות של השנה האחרונה

האילוצים שהתוכנית בודקת:

(1 סך כמות המנות שההזמנה מכילה היא לפחות 2

(2 מחיר סך ההזמנות גדול מ-200 שקלים

התוכנית מציגה את הלקוחות שעמדו בתנאים ונכנסו לשלב הסופי, לאחר מכן התוכנית מגרילה מספר אקראי עבור אחת ההזמנות שנכנסו להגרלה כך שאותה הזמנה היא הזוכה.

```
create or replace procedure TryToWin (TheDate out date,TheWinner out
number)
is
  --declare the variables:

  ThePrice number;

  i number;
  rand number;

  DishCount number;
  numParticipants number;

  cursor allOrders is
  select date_order,customer.phone,fname,lname,id_order
  from invite join customer
  on invite.phone=customer.phone
  where date_order between to_date('01-01-2020','dd-mm-yyyy')
                        and to_date('31-12-2020','dd-mm-yyyy')
  order by id_order;
  --record type of exist table
  orderRec allOrders%rowtype;

begin

  dbms_output.put_line('_____');
  dbms_output.put_line('Welcome to the lottery');
  dbms_output.put_line('_____');

  select count(phone) into numParticipants
  from customer;
  --Check that the customers existing
  if(SQL%found)
  then
    dbms_output.put_line('The total number of participants in
                        the lottery is: '|numParticipants);
    dbms_output.put_line(' * * *');
    dbms_output.put_line('The lottery participants are:');
    dbms_output.put_line('');

    TheDate:=sysdate;
    i:=0;
    --for evrey record (id_order) it takes the amount of dishes
    for orderRec in allOrders
    loop
      select sum(amount) into DishCount
      from containing
      where id_order=orderRec.Id_Order;
      --if the total amount>1 do-
      if(DishCount>=2)
      then
        --take the total sum
        select sum(price) into ThePrice
        from payment
        where id_order=orderRec.Id_Order;
        --if the price>200 do-
        if(ThePrice>200)
        then
```

```

--print the customers who entered the final
stage of the lottery
i:=i+1;
dbms_output.put_line(i||'The order number'||
' '||orderRec.Id_Order||' ' || 'of customer:');
dbms_output.put_line(' 0'||orderRec.Phone||
' '|| orderRec.Lname||' '|| orderRec.Fname);
dbms_output.put_line('');

end if;
end if;
end loop;
--select random num between 1 to the num of customers
select round(dbms_random.value(1,i)) rnum
into rand
from dual;
--print the num of winner
dbms_output.put_line('_____');
dbms_output.put_line(' Congratulations to the winner  ');
dbms_output.put_line('          * * *');
dbms_output.put_line(' The winner is lottery
candidate number:'||' '|| rand);

TheWinner:=rand;

end if;
end TryToWin

```

הרצה:

Test script	DBMS Output	Statistics	Profiler	Trace
Clear	Buffer size 10000	<input checked="" type="checkbox"/> Enabled		
<pre> Welcome to the lottery The total number of participants in the lottery is: 10 * * * The lottery participants are: 1)The order number 11 of customer: 0508126483 levi yonatan 2)The order number 5 of customer: 0504168272 amitay gali 3)The order number 6 of customer: 0524165985 cohen rona 4)The order number 9 of customer: 0541652973 shem-tov nir Congratulations to the winner * * * The winner is lottery candidate number: 3 </pre>				

Test script	DBMS Output	Statistics	Profiler	Trace												
<pre> 1 begin 2 -- Call the procedure 3 trytowin(thedate => :thedate, 4 thewinner => :thewinner); 5 end; </pre>																
<table border="1"> <thead> <tr> <th></th> <th>Variable</th> <th>Type</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/></td> <td>thedate</td> <td>Date</td> <td>09/06/2020 16:46:01</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>thewinner</td> <td>Float</td> <td>3</td> </tr> </tbody> </table>						Variable	Type	Value	<input checked="" type="checkbox"/>	thedate	Date	09/06/2020 16:46:01	<input checked="" type="checkbox"/>	thewinner	Float	3
	Variable	Type	Value													
<input checked="" type="checkbox"/>	thedate	Date	09/06/2020 16:46:01													
<input checked="" type="checkbox"/>	thewinner	Float	3													

משימה 11 – פונקציות עם חריגות

פונקציה 1

מטרת הפונקציה לאפשר לבצע הזמנה מהמסעדה מאתר האינטרנט ובלי להגיע פיזית למסעדה.

הפונקציה מקבלת כפרמטרים את מספר חבר המועדון של הלקוח ואת המנה שהלקוח רוצה להזמין.

על מנת שההזמנה תתקבל האילוצים הבאים צריכים להתקיים:

1. הלקוח קיים במערכת

2. המנה קיימת בתפריט

כשההזמנה מתקבלת יוצג ללקוח את הפרטים שלו וכן את פרוט ההזמנה. כל קלט שהמשתמש יכניס ויהיה שגוי תוצג לו הודעה שגיאה מתאימה.

הפונקציה מחזירה את מספר ההזמנה.

```

create or replace function Orders(NumMem in number,NameDish in
varchar2,DateOrder in date) return integer
is
    YourIdOrder integer;

    TheMember number;
    TheDish varchar2(15);
    TheDate date:=sysdate;
    ThePrice number;
    flag number:=1;
    rand varchar2(15);
    strLN varchar2(15) NOT NULL :='ln';
    strFN varchar2(15) NOT NULL :='fn';
    ThePhone varchar2(15);

    cursor allMembers (NumMemParmter in integer) is
    select NumMem
    from customer
    where club_member=NumMemParmter;

    cursor allDish (NameDishParmter in varchar2) is
    select NameDish
    from dish
    where name=NameDishParmter;

    cursor allOrders is
    select id_order
    from payment;

    orderRec allOrders%rowtype;

begin

    open allMembers (NumMem);
    fetch allMembers into TheMember;
    if (allMembers%notfound)
    then
        raise_application_error(-20020,'The club num member does not
exist');
    end if;
    close allMembers;

    select lname,fname,phone into strLN,strFN,ThePhone
    from customer
    where club_member=NumMem;

    dbms_output.put_line('The name of customer is:' || strLN || ' ' ||
strFN );
    dbms_output.put_line('The num phone is:' || ThePhone );
    dbms_output.put_line('The club num member is:' || NumMem );

    open allDish (NameDish);
    fetch allDish into TheDish;
    if (allDish%notfound)
    then
        raise_application_error(-20021,'The dish does not exist');
    end if;
    close allDish;

```

```

if (DateOrder<>TheDate)
then
    raise_application_error(-20022,'You need to enter the date of
today');
end if;

select price into ThePrice
from dish
where name=NameDish;

loop

    select round(dbms_random.value(1,1000)) rnum
    into rand
    from dual;

    if(rand=orderRec.Id_Order)
    then flag:=0;
    end if;
    exit when flag=1;
end loop;

insert into invite
(id_order,date_order,type_order,feedback,id_waiter,phone)
values (rand,TheDate,'take away',5,'203432117',ThePhone);

insert into payment(date_pay, price,id_order)
values (TheDate,ThePrice,rand);

YourIdOrder:=rand;
dbms_output.put_line('_____');
dbms_output.put_line('Your order is:');
dbms_output.new_line;
dbms_output.put_line('  Name dish | Price');
dbms_output.put_line('  |||NameDish|||  ||| ThePrice);

return (YourIdOrder);

end Orders;

```

דוגמאות הרצה:

דוגמא 1

משתמש הכניס לקוח שאינו חבר מועדון במסעדה.

<input type="checkbox"/>	Variable	Type	Value
<input checked="" type="checkbox"/>	result	Float	
<input checked="" type="checkbox"/>	nummem	Float	3455
<input checked="" type="checkbox"/>	namedish	String	pizza
<input checked="" type="checkbox"/>	*		

1:1
sshloimi@labdbwin
ORA-20020: The club num member does not exist

ניתן לראות שהתוכנית לא רצה וקפצה הודעה למשתמש עם מס' השגיאה ומה השגיאה.

דוגמא 2

משתמש הכניס לקוח שקיים במאגר אך מנה שלא קיימת בתפריט.
















<input type="checkbox"/>	Variable	Type	Value
<input checked="" type="checkbox"/>	result	Float	
<input checked="" type="checkbox"/>	nummem	Float	54
<input checked="" type="checkbox"/>	namedish	String	banana
<input checked="" type="checkbox"/>	*		

1:1 sshlomi@labdbwin → ORA-20021: The dish does not exist

ניתן לראות שהתוכנית לא רצה וקפצה הודעה למשתמש עם מס' השגיאה ומה השגיאה.

דוגמא 3

המשתמש הכניס מס' לקוח שקיים במאגר וכן מנה שקיימת בתפריט. הפונקציה מחזירה את מס' הזמנה.

Test script	✓ DBMS Output	Statistics	Profiler	Trace
				
				
				
1	begin			
2	-- Call the function			
3	:result := orders(nummem => :nummem,			
4	namedish => :namedish);			
5	end;			

<input type="checkbox"/>	Variable	Type	Value
<input checked="" type="checkbox"/>	result	Float	173
<input checked="" type="checkbox"/>	nummem	Float	898
<input checked="" type="checkbox"/>	namedish	String	pizza

הפלט שמוצג למשתמש -

Test script	DBMS Output	Statistics	Profiler	Trace
<div>Clear Buffer size 10000 <input checked="" type="checkbox"/> Enabled</div> <div>The name of customer is:shalom michal The num phone is:506561604 The club num member is:898</div> <div>-----</div> <div>Your order is:</div> <div><div>Name dish Price</div><div>pizza 40</div></div>				

פונקציה 2

מטרת הפונקציה – בדיקת יעילותם של המלצרים .

הפונקציה מקבלת כפרמטרים את ת.ז של המלצר וכן את מספר ההזמנות שמנהל המסעדה משער שצריך להיות לאותו מלצר לפי כמות שעות עבודה/משמרות בשבוע.

על מנת שההזמנה תתקבל האילוצים הבאים צריכים להתקיים :

1. המלצר קיים במערכת

2. מספר הזמנות גדול מ-1

הפונקציה בודקת אם מספר ההזמנות עבור אותו מלצר >2 אם כן המלצר יקבל תוספת בשכר. אחרת, בעל המסעדה יקבל הודעה בהתאם.

כל קלט שהמשתמש יכניס ויהיה שגוי תוצג לו הודעה שגיאה מתאימה.

הפונקציה מחזירה את המשכורת הסופית עבור המלצר.

```

        create or replace function WaiterSalery(IdWaiter in
varchar2,AmountOrder in number) return number
is
    --declare error #1
    waiter_invalid exception;
    pragma exception_init(waiter_invalid,-20202);
    --declare error #2
    error_input exception;
    pragma exception_init(error_input,-20203);

    --cursor for use if the waiter does exists
    cursor allWaiter (Parmter in varchar2) is
    select IdWaiter
    from waiter
    where id_waiter=Parmter;

    FunctionResult number;
    countOrder number;
    NewSalery number;
    Thewaiter varchar2(15);
    strLN varchar2(15) NOT NULL := 'ln';
    strFN varchar2(15) NOT NULL := 'fn';
    ThePhone varchar2(15);

begin
    --Check if the waiter exists
    open allWaiter (IdWaiter);
    fetch allWaiter into Thewaiter;
    if (allWaiter%notfound)
    then
        --if the waiter does not exist he throws us to declare #1
        raise waiter_invalid;
    end if;
    close allWaiter;
    --select the details of the waiter
    select lname,fname into strLN,strFN
    from waiter
    where id_waiter=IdWaiter;

    select phone into ThePhone
    from waiter_phone
    where id_waiter=IdWaiter;
    --print the details
    dbms_output.put_line('The name of waiter is:' || strLN || ' ' ||
strFN );
    dbms_output.put_line('The phone number is:0' || thephone );

    --if the parameter that the user entered<1 he throws him to declare
#2
    if(AmountOrder<1)
    then
        raise error_input;
    end if;

    --select the num of orders of last year
    select count(id_order) into countOrder
    from invite
    where(id_waiter=IdWaiter) and (date_order between to_date('01-01-
2020','dd-mm-yyyy')
                                and to_date('31-12-
2020','dd-mm-yyyy'));

```


דוגמא 2

עבור מלצר בעל ת.ז. 216887584 שהוא בעל 3 הזמנות, הפונקציה תחזיר את המשכורת שלו בתוספת בונוס וכן את פרטיו של המלצר.

<input type="checkbox"/>	Variable	Type	Value
<input checked="" type="checkbox"/>	result	Float	8250
<input checked="" type="checkbox"/>	idwaiter	String	216887584
<input checked="" type="checkbox"/>	amountorder	Float	2
<input checked="" type="checkbox"/>	*		

1:1 sshlomi@labdbwin Executed in 0.008 seconds

Test script DBMS Output Statistics Profiler Trace

Clear Buffer size 10000 ☒ Enabled

```
The name of waiter is:dadon dor
The phone number is:0549659999
The num of orders is: 3
```

דוגמא 3

עבור מלצר שלא קיים במערכת, תוצג שגיאה מס' 20200 עם הודעה בהתאם.

<input type="checkbox"/>	Variable	Type	Value
<input checked="" type="checkbox"/>	result	Float	
<input checked="" type="checkbox"/>	idwaiter	String	445533226
<input checked="" type="checkbox"/>	amountorder	Float	2
<input checked="" type="checkbox"/>	*		

1:1 sshlomi@labdbwin ORA-20200: the waiter does not exist

דוגמא 4

עבור מס' הזמנות לא הגיוני, (לא ייתכן שמלצר לא עבד בכלל) תוצג שגיאה מס' 20201 עם הודעה בהתאם.

<input type="checkbox"/>	Variable	Type	Value
<input checked="" type="checkbox"/>	result	Float	
<input checked="" type="checkbox"/>	idwaiter	String	216887584
<input checked="" type="checkbox"/>	amountorder	Float	0
<input checked="" type="checkbox"/>	*		

1:1 sshlomi@labdbwin ORA-20201: the amount of orders should be more then 1

משימה 12 – טריגרים

טריגר 1

קוד PL/SQL

```
create or replace trigger NewCustomer
before insert
on customer
for each row

declare

cursor allCustomer is
select phone
from customer;

phoneRec allCustomer%rowtype;

begin

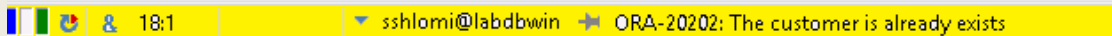
for phoneRec in allcustomer
loop
if (phoneRec.Phone=:new.phone)
then
raise_application_error(-20202,'The customer is already exists');
end if;
end loop;

end NewCustomer;
```

דוגמאות הרצה:

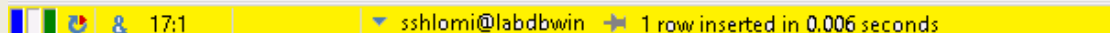
1 דוגמא

הכנסת לקוח שכבר קיים במערכת-
תוצג שגיאה מס' 20202 עם הודעה בהתאם.

SQL	Output	Statistics
<pre>insert into customer (lname, fname, phone, email, club_member, address) values('shalom','michal',0506561604,'mic@gmail.com',898,'89 derech amelech');</pre>		
		

2 דוגמא

הכנסת לקוח שלא קיים במערכת-
ההכנסה תתבצע בהצלחה.

SQL	Output	Statistics
<pre>insert into customer (lname, fname, phone, email, club_member, address) values('leci','nati',0505361606,'nati@gmail.com',454,'89 derech amelech');</pre>		
		

טריגר 2

קוד PL/SQL

```
create or replace trigger waiter
before update
on waiter
for each row

declare

countOrders number;

begin

select count(id_order) into countOrders
from invite
where id_waiter=:old.id_waiter;

if(countOrders<2)
then
raise_application_error(-20202,'This waiter order number is less than 2');
end if;

end waiter;
```

דוגמאות הרצה:

דוגמא 1

תוספת בשכר עבור מלצר מס' 216887584 בוצעה בהצלחה.

ואחרי העדכון:

```
update waiter
set salary=salary+200
where id_waiter='216887584';
```

	ID_WAITER	DOB	FNAME	LNAME	SALARY
▶ 1	203432117	01/02/1997	shir	levi	9000
2	433432117	02/08/1987	elroyi	cohen	4900
3	243432817	08/08/1996	tamar	somech	6600
4	261792117	09/05/1990	lea	levi	6500
5	203487217	07/06/2000	moti	dan	7000
6	303432117	05/11/2002	miri	haron	5690
7	203436948	08/10/1993	nir	levi	5500
8	947232117	18/05/1998	shira	noach	6000
9	264987515	10/12/2005	moshe	cohen	9100
10	216887584	10/05/2010	dor	dadon	8200

ניתן לראות את רשומה 10 לפני העדכון:

	ID_WAITER	DOB	FNAME	LNAME	SALARY
▶ 1	203432117	01/02/1997	shir	levi	9000
2	433432117	02/08/1987	elroyi	cohen	4900
3	243432817	08/08/1996	tamar	somech	6600
4	261792117	09/05/1990	lea	levi	6500
5	203487217	07/06/2000	moti	dan	7000
6	303432117	05/11/2002	miri	haron	5690
7	203436948	08/10/1993	nir	levi	5500
8	947232117	18/05/1998	shira	noach	6000
9	264987515	10/12/2005	moshe	cohen	9100
10	216887584	10/05/2010	dor	dadon	8000

דוגמא 2

מלצר מס' 243432817 לא קיבל תוספת בשכר ותוצג מס' שגיאה 20202 עם הודעה בהתאם.

SQL Output Statistics

```
update waiter
set salary=salary+200
where id_waiter='243432817';
```

18:29 sshlomi@labdbwin → ORA-20202: This waiter order number is less than 2

שלב ג' – אינטגרציה

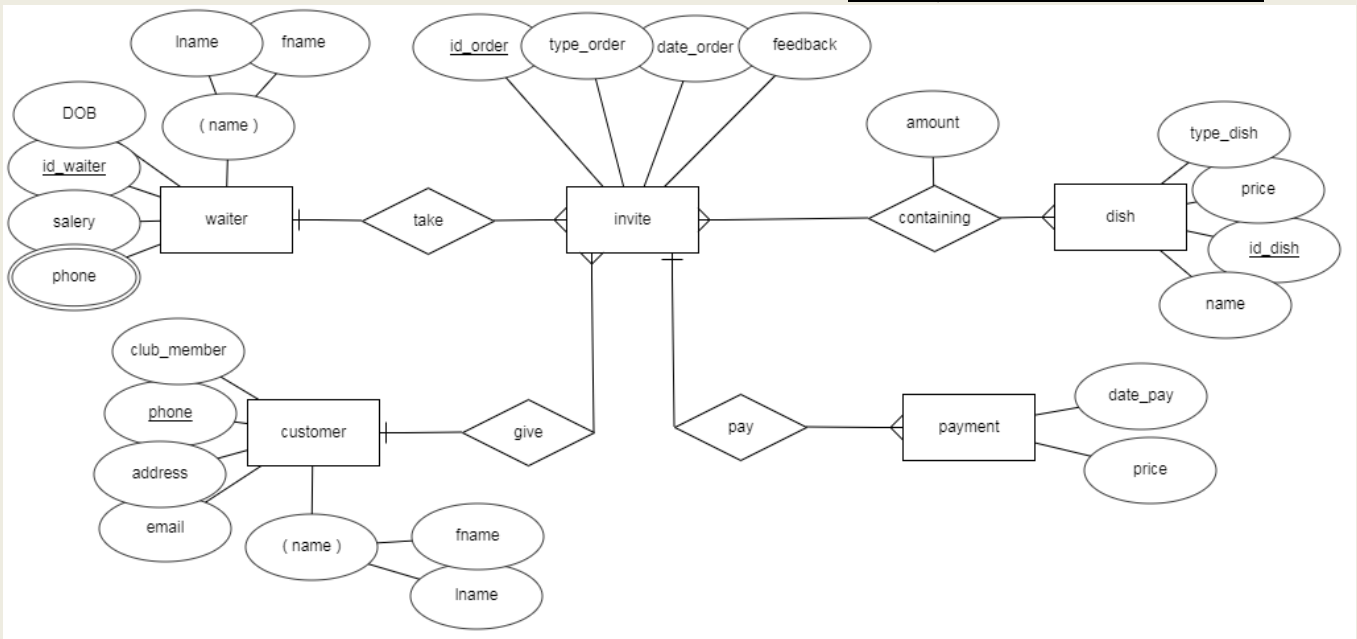
שלב 1 - מעבר מסכמה CREATE TABLE לתרשים ERD

- כל טבלה הופכת לישות
- כל עמודה הופכת לתכונה
- עמודה בטבלה מסתעפת, (ז"א טבלה שנוצרה בעקבות זאת שהיא מקשרת בין 2 טבלאות אחרות-היא הופכת קשר של רבים לרבים ל-2 קשרים של רבים ליחיד)
טבלה כזאת שהמפתחות הזרים שלה הם מפתחות ראשיים-העמודה הופכת לתכונה על הקשר.
- טבלה עם מפתח ראשי משלה ומפתח זר שמסומן כראשי - טבלה שהופכת לישות חלשה שמחוברת ע"י קשר חלש לישות החזקה . המפתח הראשי הינו מפתח חלקי של הישות החלשה והמפתח הזר הינו המפתח של הישות החזקה.

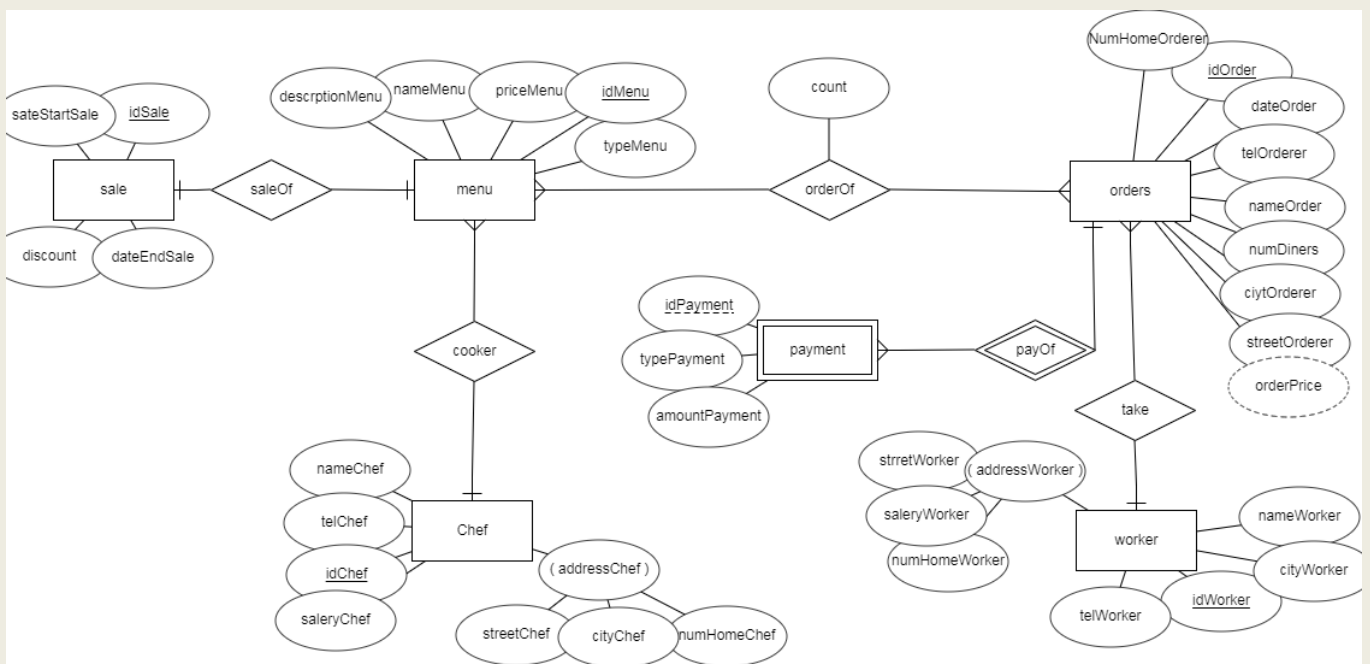
קביעת הקשרים :

- קשר של יחיד לרבים- טבלת הרבים מקבלת את המפתח הראשי של טבלת היחיד ומפתח זה הופך להיות מפתח זר בטבלת הרבים. טבלה שיש בה מפתח זר יש לה קשר של יחיד לרבים עם הטבלה שאצלה אותו מפתח הוא מפתח ראשי.
- קשר של רבים לרבים- טבלה שמחוברת בין 2 טבלאות אחרות והמפתחות הראשים של אותן טבלאות הם המפתחות הראשיים שלה וגם הזרים שלה.

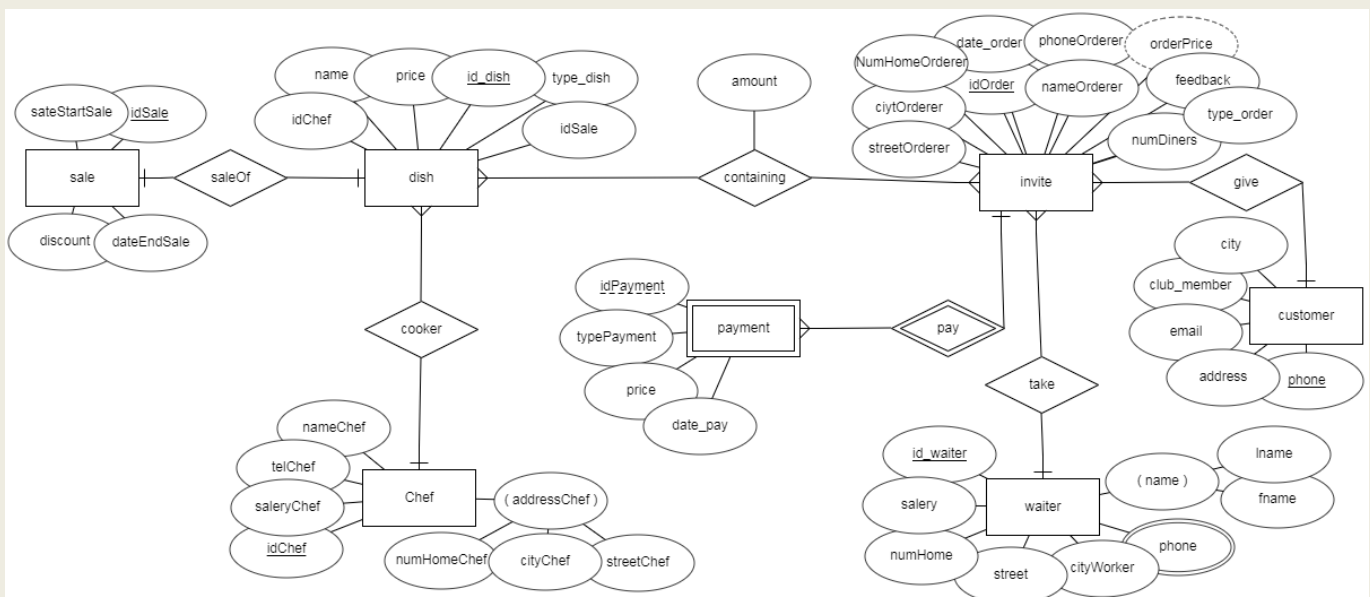
תרשים ERD של הפרויקט שלי



תרשים ERD של פרויקט אחר



תרשים ERD INTAGRATED



My Create Table

```
CREATE TABLE waiter
(
  id_waiter VARCHAR2(10) NOT NULL,
  DOB DATE NOT NULL,
  fname VARCHAR2(20) NOT NULL,
  lname VARCHAR2(20) NOT NULL,
  salery NUMBER(10) NOT NULL,
  PRIMARY KEY (id_waiter)
);

CREATE TABLE customer
(
  lname VARCHAR2(20) NOT NULL,
  fname VARCHAR2(20) NOT NULL,
  phone NUMBER(10) NOT NULL,
  email VARCHAR2(20) NOT NULL,
  club_member NUMBER(10) NOT NULL,
  address VARCHAR2(20) NOT NULL,
  PRIMARY KEY (phone)
);

CREATE TABLE dish
(
  id_dish VARCHAR2(10) NOT NULL,
  price NUMBER(10) NOT NULL,
  name VARCHAR2(20) NOT NULL,
  type_dish VARCHAR(20) NOT NULL,
  PRIMARY KEY (id_dish)
);

CREATE TABLE waiter_phone
(
  phone NUMBER(10) NOT NULL,
  id_waiter VARCHAR2(10) NOT NULL,
  PRIMARY KEY (phone),
  FOREIGN KEY (id_waiter) REFERENCES waiter(id_waiter)
  on delete cascade
);

CREATE TABLE invite
(
  id_order VARCHAR2(10) NOT NULL,
  date_order DATE NOT NULL,
  type_order VARCHAR2(20) NOT NULL,
  feedback VARCHAR2(20) NOT NULL,
  id_waiter VARCHAR2(10) NOT NULL,
  phone NUMBER(10) NOT NULL,
  PRIMARY KEY (id_order),
  FOREIGN KEY (id_waiter) REFERENCES waiter(id_waiter)
  on delete cascade,
  FOREIGN KEY (phone) REFERENCES customer(phone)
  on delete cascade
);

CREATE TABLE payment
(
  date_pay DATE NOT NULL,
  price NUMBER(10) NOT NULL,
  id_order VARCHAR2(10) NOT NULL,
  FOREIGN KEY (id_order) REFERENCES invite(id_order )
  on delete cascade
);

CREATE TABLE containing
(
  amount NUMBER (20) NOT NULL,
  id_order VARCHAR2(10) NOT NULL,
  id_dish VARCHAR2(10) NOT NULL,
  PRIMARY KEY (id_order, id_dish),
  FOREIGN KEY (id_order) REFERENCES invite(id_order)
  on delete cascade,
  FOREIGN KEY (id_dish) REFERENCES dish(id_dish)
  on delete cascade
);
```

The Integrated Scheme

```
--מלצרים לטבלת עמודות הוספת--

alter table waiter
add (city varchar2(15) not null,
     numHome number not null,
     street varchar2(15) not null);

--שף טבלת יצירת--

CREATE TABLE chef
(
  idChef INT NOT NULL,
  nameChef varchar(10) NOT NULL,
  telChef INT NOT NULL,
  cityChef varchar (10) NOT NULL,
  streetChef varchar (10) NOT NULL,
  numHomeChef INT NOT NULL,
  salaryChef INT NOT NULL,
  PRIMARY KEY (idChef));

--חדשה טבלה יצירת--

CREATE TABLE sale
(
  idSale INT NOT NULL,
  discount INT NOT NULL,
  dateStartSale date NOT NULL,
  dateEndSale date NOT NULL,
  PRIMARY KEY (idSale));

--הזמנות לטבלת עמודות הוספת--

alter table invite
add (nameOrderer varchar(10) NOT NULL,
     numDiners INT,
     cityOrderer varchar(15),
     addressOrderer varchar(15));

--הזמנות בטבלת עמודה שם שינוי--

alter table invite
rename column phone to phoneOrderer;

--תשלום לטבלת עמודות הוספת--

alter table payment
add (id_payment INT NOT NULL,
     PRIMARY KEY (id_payment),
     typePayment varchar(10) NOT NULL);

--מנות לטבלת עמודות הוספת--

alter table dish
add(idChef INT,
    idSale INT,
    FOREIGN KEY (idChef) REFERENCES chef(idChef),
    FOREIGN KEY (idSale) REFERENCES sale(idSale));

--לקוחות לטבלת עמודה הוספת--

alter table customer
add city varchar2(15) NOT NULL;
```

שלב 3 - הכנסת נתונים לבסיס הנתונים המשולב

טבלת CHEF

CHEF				
Owner	Table	Number of records		
SSHLOMI	CHEF	30		
Name	Type	Size	Data	
IDCHEF	NUMBER		Sequence(Start, [Inc], [WithinParent])	
NAMECHEF	VARCHAR2	10	FirstName	
TELCHEF	NUMBER		['05'00000000]	
CITYCHEF	VARCHAR2	10	City	
STREETCHEF	VARCHAR2	10	List('AKIVA', 'BEGIN', 'ETROG', 'AFUNA', 'BEN GURYON', 'HSACHAR', 'NETAN')	
NUMHOMECHEF	NUMBER		Random(1,190)	
SALARYCHEF	NUMBER		Random(5000, 9500)	

אישור הכנסה-

sshloimi@labdbwin 30 rows selected in 0.087 seconds

טבלת WAITER

WAITER				
Owner	Table	Number of records		
SSHLOMI	WAITER	30		
Name	Type	Size	Data	
ID_WAITER	VARCHAR2	10	[0000000000]	
DOB	DATE		Random(01-01-1950,01-01-2000)	
FNAME	VARCHAR2	20	FirstName	
LNAME	VARCHAR2	20	LastName	
SALARY	NUMBER	10	Random(5000, 9500)	
CITY	VARCHAR2	15	City	
NUMHOME	NUMBER		Random(1, 110)	
STREET	VARCHAR2	15	List('ETROG', 'AMELECH', 'BEGIN', 'BEN GURYON', 'APALMACH', 'DAYYAN', 'SHPIRA')	

אישור הכנסה-

sshloimi@labdbwin 30 rows selected in 0.076 seconds

טבלת WAITER PHONE

WAITER_PHONE				
Owner	Table	Number of records		
SSHLOMI	WAITER_PHONE	30		
Name	Type	Size	Data	
PHONE	NUMBER	10	['05'00000000]	
ID_WAITER	VARCHAR2	10	List(select id_waiter from waiter)	

אישור הכנסה-

sshloimi@labdbwin 30 rows selected in 0.105 seconds

טבלת CUSTOMER

CUSTOMER				
Owner	Table	Number of records		
SSHLOMI	CUSTOMER	30		
Name	Type	Size		Data
LNAME	VARCHAR2	20	▲	LastName
FNAME	VARCHAR2	20	▲	FirstName
PHONE	NUMBER	10	▲	['05'00000000]
EMAIL	VARCHAR2	20	▲	Email
CLUB_MEMBER	NUMBER	10	▲	Random(10, 100)
ADDRESS	VARCHAR2	20	▲	Address1

אישור הכנסה-

sshloimi@labdbwin 30 rows selected in 0.117 seconds

טבלת SALE

SALE				
Owner	Table	Number of records		
SSHLOMI	SALE	30		
Name	Type	Size		Data
IDSALE	NUMBER		▲	Sequence(Start, [Inc], [WithinParent])
DISCOUNT	NUMBER		▲	Random(0,30)
DATESTARTSALE	DATE		▲	Random(01-01-2010, 01-06-2020)
DATEENDSALE	DATE		▲	Random(01-01-2010, 01-06-2020)

אישור הכנסה-

sshloimi@labdbwin 30 rows selected in 0.134 seconds

טבלת DISH

DISH				
Owner	Table	Number of records		
SSHLOMI	DISH	30		
Name	Type	Size		Data
ID_DISH	VARCHAR2	10	▲	Sequence(Start, [Inc], [WithinParent])
PRICE	NUMBER	10	▲	Random(10, 60)
NAME	VARCHAR2	20	▲	Elements.Name
TYPE_DISH	VARCHAR2	20	▲	List('firsn dish', 'main dish', 'last dish')
IDCHEF	NUMBER		▲	List(select IDCHEF from chef)
IDSALE	NUMBER		▲	List(select IDSALE from sale)

אישור הכנסה-

sshloimi@labdbwin 30 rows selected in 0.146 seconds

טבלת INVITE

INVITE				
Owner		Table		Number of records
SSHLOMI		INVITE		30
Name	Type	Size	Data	
ID_ORDER	VARCHAR2	10	Sequence(Start, [Inc], [WithinParent])	
DATE_ORDER	DATE		Random(01-01-2010, 01-06-2020)	
TYPE_ORDER	VARCHAR2	20	List('sitting', 'take away')	
FEEDBACK	VARCHAR2	20	Random(1, 5)	
ID_WAITER	VARCHAR2	10	List(select id_waiter from waiter)	
PHONEORDERER	NUMBER	10	List(select phone from customer)	
NAMEORDERER	VARCHAR2	10	List(select fname from customer)	
NUMDINERS	NUMBER		Random(1, 10)	
CITYORDERER	VARCHAR2	15	List(select city from customer)	
ADDRESSORDERER	VARCHAR2	15	List(select address from customer)	

אישור הכנסה-

sshloimi@labdbwin → 30 rows selected in 0.158 seconds

טבלת PAYMENT

PAYMENT				
Owner		Table		Number of records
SSHLOMI		PAYMENT		30
Name	Type	Size	Data	
DATE_PAY	DATE		List(select date_order from invite)	
PRICE	NUMBER	10	Random(40, 800)	
ID_ORDER	VARCHAR2	10	List(select id_order from invite)	
ID_PAYMENT	NUMBER		Sequence(Start, [Inc], [WithinParent])	
TYPEPAYMENT	VARCHAR2	10	List('card', 'cash')	

אישור הכנסה-

sshloimi@labdbwin → 30 rows selected in 0.175 seconds

טבלת CONTAINING

CONTAINING				
Owner		Table		Number of records
SSHLOMI		CONTAINING		30
Name	Type	Size	Data	
AMOUNT	NUMBER	20	Random(1, 4)	
ID_ORDER	VARCHAR2	10	List(select id_order from invite)	
ID_DISH	VARCHAR2	10	List(select id_dish from dish)	

אישור הכנסה-

sshloimi@labdbwin → 30 rows selected in 0.063 seconds

שלב 4 – הרצת שאילתות

דוגמאות הרצת שאילתות מהפרויקט שלי על בסיס הנתונים המשותף:

***** כל הסימונים בצהוב אלו השינויים/תיקונים שהייתי צריכה לעשות בשאילתה על**

מנת שהיא תרוץ ותעבוד באופן חלק.***

דוגמא 1

שאילתת select מקור מס' 3 : הצגת פרטי לקוח ששילם מעל 100 שקלים עבור הזמנה.

```
select invite.id_order , price ,customer.phone , lname, fname
from invite join customer
on invite.phoneorderer = customer.phone
      join payment
on invite.id_order = payment.id_order
where (price>100)
order by price
```

	ID_ORDER	PRICE	PHONE	LNAME	FNAME
1	29	159	507076618	Travolta	Elijah
2	16	223	549336097	Henstridge	Jeroen
3	11	257	547814021	Diehl	Olga
4	8	302	522425550	Pressly	Buddy
5	21	365	547814021	Diehl	Olga
6	27	458	555629080	Teng	Gin
7	16	470	549336097	Henstridge	Jeroen
8	25	588	577786644	Tankard	Spencer
9	8	598	522425550	Pressly	Buddy
10	29	678	507076618	Travolta	Elijah
11	23	688	507076618	Travolta	Elijah
12	27	707	555629080	Teng	Gin
13	26	764	569151342	Jones	Hal

דוגמא 2

שאילתת select מקור מס' 4 : השאילתה מחזירה את העובדים שביצעו יותר מהזמנה אחת בחודש.

השאילתה שמחזירה את שם העובד, מס' הזמנות שלו ותאריך אחרון עבור ההזמנה שביצע .

```
select lname,
       fname,
       waiter.id_waiter,
       count(id_order) as "total order",
       max(date_order) as "last order"
from waiter join invite
on waiter.id_waiter = invite.id_waiter
group by waiter.lname , waiter.fname ,waiter.id_waiter
having (count(id_order)>1)
```

	LNAME	FNAME	ID_WAITER	total order	last order
1	Bugnon	Miko	368724567	2	27/03/2017
2	Springfield	Judi	615198703	3	10/06/2016
3	Kenoly	Tia	254051834	2	24/09/2016
4	Palin	Christmas	430372296	2	12/12/2017
5	McLean	Oliver	708443125	2	14/02/2020
6	Connery	Regina	057159451	2	09/08/2016
7	Chesnutt	Daniel	278351439	2	02/08/2011
8	Shaw	Hope	999535498	2	16/12/2013
9	Haysbert	Wes	906700509	3	17/12/2019
10	Skarsgard	Alicia	229970772	3	28/09/2018

דוגמא 3

מבט מקור מס' 2 : טבלה המשקפת את כל הדברים שקשורים בכסף. (ז"א הוצאות והכנסות)

```
create view profit_check as
select w.id_waiter,
       w.dob,
       w.fname,
       w.lname,
       w.salary,
       i.id_order,
       i.date_order,
       i.type_order,
       i.feedback,
       i.phoneorderer,
       p.date_pay,
       p.price
from waiter w, payment p, invite i
where w.id_waiter=i.id_waiter
and i.id_order=p.id_order
```

ניתן לראות שה- VIWE נוצר בהצלחה לאחר השינוי .

SQL Output Statistics												
select * from profit_check												
	ID_WAITER	DOB	FNAME	LNAME	SALARY	ID_ORDER	DATE_ORDER	TYPE_ORDER	FEEDBACK	PHONEORDERER	DATE_PAY	PRICE
1	052942545	01/08/1994	Henry	Jones	6693	27	10/11/2020	sitting	2	555629080	05/06/2016	707
2	658800670	05/04/2003	Ricardo	Fiennes	9735	16	08/10/2017	sitting	4	549936097	18/11/2012	223
3	846865895	02/01/2001	Randy	Giannini	7414	26	15/04/2019	sitting	4	569151342	15/04/2019	764
4	382937090	23/04/2006	Percy	Durning	8949	29	24/07/2020	take away	4	507076618	18/12/2015	159
5	154366828	09/10/2007	Tony	Curry	9944	8	03/04/2017	sitting	5	522425550	19/02/2011	598
6	312790783	24/01/1980	Tim	Sanders	6413	25	06/05/2017	sitting	1	577786644	11/09/2014	588
7	696794989	25/02/1970	Penelope	Pearce	6554	11	22/05/2017	sitting	2	547814021	10/12/2014	257
8	052942545	01/08/1994	Henry	Jones	6693	27	10/11/2020	sitting	2	555629080	29/06/2020	458
9	363621177	14/05/1984	Ellen	Cattrall	6210	21	17/10/2020	take away	2	547814021	28/04/2014	89
10	382937090	23/04/2006	Percy	Durning	8949	29	24/07/2020	take away	4	507076618	18/12/2015	678
11	154366828	09/10/2007	Tony	Curry	9944	8	03/04/2017	sitting	5	522425550	24/07/2020	302
12	363621177	14/05/1984	Ellen	Cattrall	6210	21	17/10/2020	take away	2	547814021	23/12/2011	365
13	307691765	08/09/1972	Juliet	Miller	6278	23	21/02/2019	sitting	5	507076618	05/06/2016	688
14	658800670	05/04/2003	Ricardo	Fiennes	9735	16	08/10/2017	sitting	4	549936097	29/06/2020	470

דוגמא 4

שאילתת delete מקור מס' 2 : מחיקת כל ההזמנות הלא רלוונטיות (ז"א שלא בשנת 2020)

```
delete
from invite
where id_order in (select id_order
                   from payment
                   where (date_pay between to_date('01-01-2010','dd-mm-yyyy')
                        and to_date('31-12-2019','dd-mm-yyyy'))
                   and payment.price>100)
```


167:1 sshlomi@labdbwin 16 rows deleted in 0.060 seconds

דוגמאות הרצת שאילתות מהפרויקט האחר על בסיס הנתונים המשותף:

***** כל הסימונים בצהוב אלו השינויים/תיקונים שהייתי צריכה לעשות בשאילתה על מנת שהיא תרוץ ותעבוד באופן חלק.*****

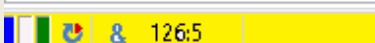
דוגמא 1

שאילתת select מקור מס' 6 : השאילתה מציגה את שם הלקוח ומספר טלפון עבור הזמנות שכמות המנות שלהן > 4.

SQL	Output	Statistics
<pre>SELECT i.nameorderer,i.phoneorderer FROM invite i WHERE id_Order IN (SELECT i.id_order FROM containing,invite where containing.id_order=invite.id_Order AND amount>4);</pre>		
		
	NAMEORDERER	PHONEORDERER
1	Patricia	577786644
2	Elijah	527783467
3	Gin	522425550
4	Kimberly	504469401
5	Cornell	547814021
6	Gin	549336097
7	Mika	547814021
8	Kimberly	507076618
9	Kyle	577786644
10	Millie	569151342
11	Lucinda	555629080
12	Temuera	507076618

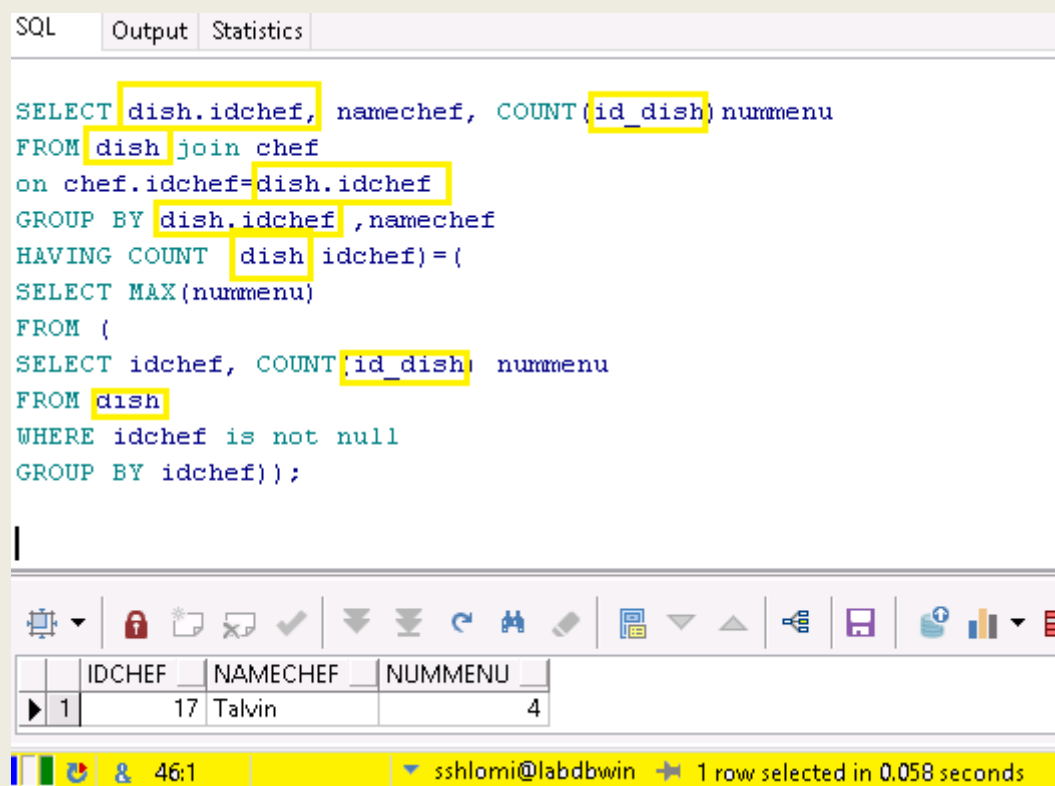
דוגמא 2

שאילתת delete מקור מס' 2 : השאילתה מוחקת את כל ההזמנות שבוצעו לפני 4 שנים מהשנה הנוכחית

SQL	Output	Statistics
<pre>delete from invite where EXTRACT(YEAR FROM date_Order)<=EXTRACT(YEAR FROM sysdate)-4; commit;</pre>		
		
sshlorini@labdbwin 17 rows deleted in 0.010 seconds		

דוגמא 3

שאלת select מקור מס' 4: השאילתה מציגה את השף שהכין הכי הרבה מנות במסעדה מבין שאר השפים. השאילתה מחזירה את מס' השף, השם ומס' המנות שלו.



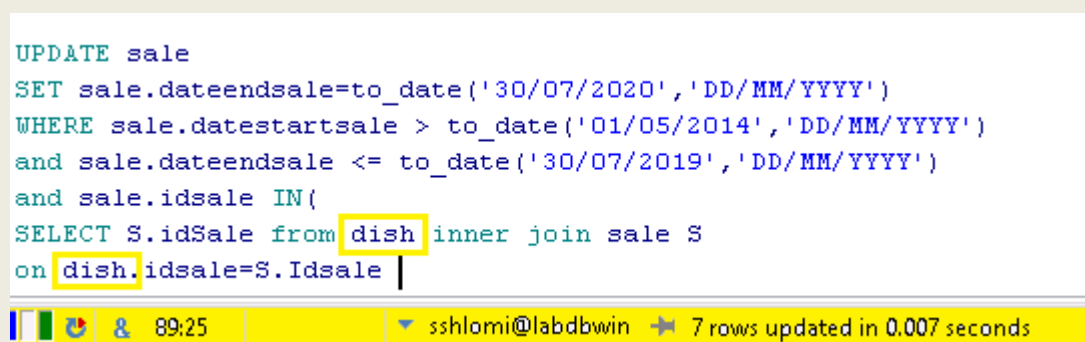
```
SQL      Output  Statistics
SELECT dish.idchef, namechef, COUNT(id_dish) nummenu
FROM dish join chef
on chef.idchef=dish.idchef
GROUP BY dish.idchef, namechef
HAVING COUNT(dish idchef)=(
SELECT MAX(nummenu)
FROM (
SELECT idchef, COUNT(id_dish) nummenu
FROM dish
WHERE idchef is not null
GROUP BY idchef));
```

	IDCHEF	NAMECHEF	NUMMENU
1	17	Talvin	4

46:1 sshlomi@labdbwin 1 row selected in 0.058 seconds

דוגמא 4

שאלת update מקור מס' 1: השאילתה מעדכנת את תאריך סיום המבצע עבור כל מוצר בטווח תאריכים שנבחר



```
UPDATE sale
SET sale.dateendsale=to_date('30/07/2020','DD/MM/YYYY')
WHERE sale.datestartsale > to_date('01/05/2014','DD/MM/YYYY')
and sale.dateendsale <= to_date('30/07/2019','DD/MM/YYYY')
and sale.idsale IN(
SELECT S.idSale from dish inner join sale S
on dish.idsale=S.Idsale )
```

89:25 sshlomi@labdbwin 7 rows updated in 0.007 seconds

דוגמא 5

שאלת select מקור מס' 2 : השאילתה מציגה את שם המלצר, מס' מלצר והמשכורת שלו עבור מלצרים שמשכורתם גבוהה ממוצע המשכורת של העובדים.

```
select w.id_waiter, w.fname, w.lname, w.salary
from waiter w
where w.salary > (select AVG(w2.salary)
                  from waiter w2);

select count(w.id_waiter) countWorker
from waiter w
where w.salary > (select AVG(w2.salary)
                  from waiter w2);
```

Select waiter Select waiter

	ID_WAITER	FNAME	LNAME	SALARY
1	509754760	David	Brooke	9939
2	981801935	France	Lyonne	9623
3	160611919	Rhett	Field	8899
4	154366828	Tony	Curry	9944
5	849383132	Rhona	Carlisle	8804
6	093057246	Peabo	Orlando	9728
7	704102509	Jill	Oszajca	8868
8	492704820	Molly	McCann	9262
9	658800670	Ricardo	Fiennes	9735
10	204761107	Rhona	Reed	9257
11	044881371	Scott	Doucette	8718
12	452152197	Edgar	Lowe	9014
13	759222447	Jonny	Stamp	8233

ssh@lomi@labdbwin 16 rows selected in 0.033 seconds

דוגמא 6

שאלת delete מקור מס' 1 : השאילתה מוחקת את ההזמנה הישנה ביותר.

SQL Output Statistics

```
delete from invite i
where i.id_order in (select i2.id_order from invite i2
                    where i2.date_order <= all(select i3.date_order
                                              from invite i3
                                              where i2.id_order <> i3.id_order));
```

ssh@lomi@labdbwin 1 row deleted in 0.017 seconds