

טלאל גינזברג

מע"ן עמר

יולדות- ניהול ארוחות ליולדות

תוכן עניינים:

מבוא	3
טבלאות erd,dsd	4
createTable	5-8
שיטות להכנסת נתונים	9-11
גיבוי שלב א'	12
queries	13-18
alter tables	19
תכנית 1	20-25
תכנית 2	26-33

מבוא:

המערכת תנהל עבור קהילות שונות מידע על יולדות ועל הארוחות שחברי קהילה אחרים מכינים להם.

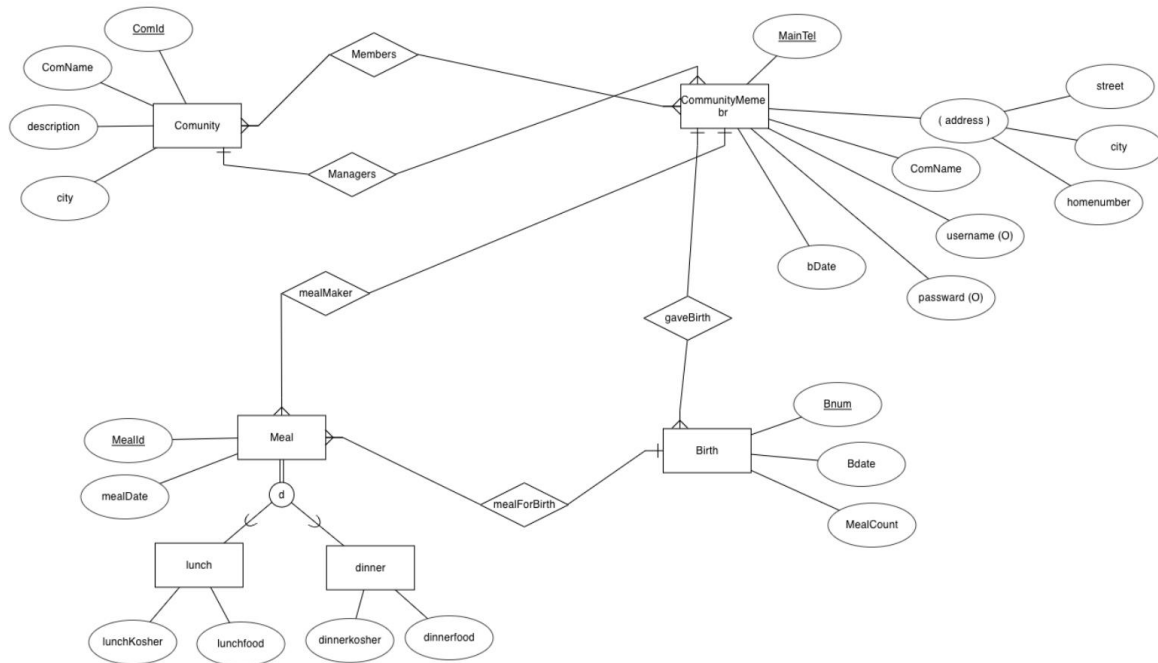
במערכת יש קהילות ובה נמצאים חברי קהילות, יולדת היא חברת קהילה וגם מבשלת היא חברת קהילה. ישנם כמה וכמה יולדות וכמה אנשים שמכינים לכל יולדת ארוחה.

כל ארוחה היא מסוג ארוחת ערב או צהריים, ואז לכל ארוחה יש לה אפשרויות שונות של מאכלים ושל כשרות(בשרי, חלבי, פרווה).

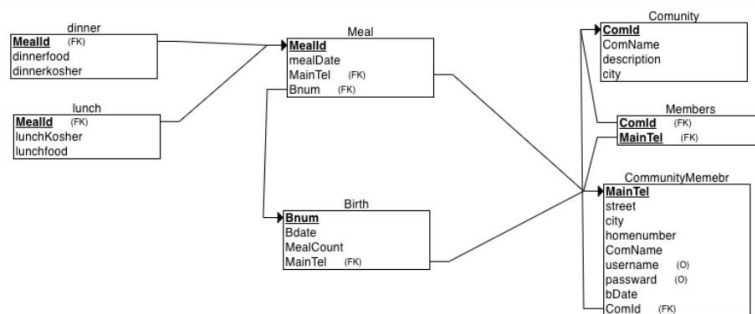
לכל קהילה יש מספר מזהה, שם קהילה, תיאור קהילה עיר, ולכל קהילה יש חבר קהילה שמנהל אותה, ובנוסף חברי קהילה רגילה.

לחבר קהילה המזהה שלו הוא מספר הטלפון שלו, הכתובת שלו, תאריך הלידה. יולדת היא סוג של חברת קהילה, עם תאריך לידה, מספר לידה, מספר אנשים עבורם צריך להכין את הארוחה.

:erd



:dsd



:createtable

עבור :community

```
CREATE TABLE Community
(
  ComId INT NOT NULL,
  ComName VARCHAR2(100) NOT NULL,
  Cdescription VARCHAR2(1000) NOT NULL,
  city VARCHAR2(50) NOT NULL,
  PRIMARY KEY (ComId)
);
```

Columns of COMMUNITY

	Name	Type	Nullable	Default	Comments
1	COMID	INTEGER			
2	COMNAME	VARCHAR2(100)			
3	CDESCRIPTION	VARCHAR2(1000)			
4	CITY	VARCHAR2(50)			

עבור :community member

```

CREATE TABLE CommunityMember
(
    MainTel VARCHAR2(10) NOT NULL,
    street VARCHAR2(50) NOT NULL,
    city VARCHAR2(50) NOT NULL,
    homenum INT NOT NULL,
    ComName VARCHAR(20) NOT NULL,
    username VARCHAR(50),
    password INT,
    ComId INT NOT NULL,
    PRIMARY KEY (MainTel),
    FOREIGN KEY (ComId) REFERENCES Community(ComId)
);

```

Columns of COMMUNITYMEMBER

	Name	Type	Nullable	Default	Comments
1	MAINTEL	VARCHAR2(10)			
2	STREET	VARCHAR2(50)			
3	CITY	VARCHAR2(50)			
4	HOMENUMBER	INTEGER			
5	COMNAME	VARCHAR2(20)			
6	USERNAME	VARCHAR2(50)	Y		
7	PASSWORD	INTEGER	Y		
8	COMID	INTEGER			

עבור birth:

```

FOREIGN KEY (ComId) REFERENCES Community(ComId)
);

CREATE TABLE Birth
(
    Bnum INT NOT NULL,
    Bdate DATE NOT NULL,
    MealCount INT NOT NULL,
    MainTel VARCHAR2(10) NOT NULL,
    PRIMARY KEY (Bnum),
    FOREIGN KEY (MainTel) REFERENCES CommunityMember(MainTel)
);

```

Columns of BIRTH					
	Name	Type	Nullable	Default	Comments
1	BNUM	INTEGER			
2	BDATE	DATE			
3	MEALCOUNT	INTEGER			
4	MAINTEL	VARCHAR2(10)			

עבור meal:

```
);
CREATE TABLE Meal
(
    MealId INT NOT NULL,
    MealDate Date NOT NULL,
    MainTel VARCHAR2(10) NOT NULL,
    Bnum INT NOT NULL,
    PRIMARY KEY (MealId),
    FOREIGN KEY (MainTel) REFERENCES CommunityMember(MainTel),
    FOREIGN KEY (Bnum) REFERENCES Birth(Bnum)
);
```

Columns of MEAL					
	Name	Type	Nullable	Default	Comments
1	MEALID	INTEGER			
2	MEALDATE	DATE			
3	MAINTEL	VARCHAR2(10)			
4	BNUM	INTEGER			

עבור dinner:

```
CREATE TABLE dinner
(
    MealId INT NOT NULL,
    DinnerFood varchar2(20) not null,
    DinnerKosher varchar2(20) not null,
    PRIMARY KEY (MealId),
    FOREIGN KEY (MealId) REFERENCES Meal(MealId)
);
```

	Name	Type	Nullable	Default	Comments
►	MEALID	INTEGER			
2	DINNERFOOD	VARCHAR2(20)			
3	DINNERKOSHER	VARCHAR2(20)			

עבור lunch:

```
CREATE TABLE lunch
(
    MealId INT NOT NULL,
    LunchFood varchar2(20) not null,
    LunchKosher  varchar2(20) not null,

    PRIMARY KEY (MealId),
    FOREIGN KEY (MealId) REFERENCES Meal(MealId)
);
```

	Name	Type	Nullable	Default	Comments
►	MEALID	INTEGER			
2	LUNCHFOOD	VARCHAR2(20)			
3	LUNCHKOSHER	VARCHAR2(20)			

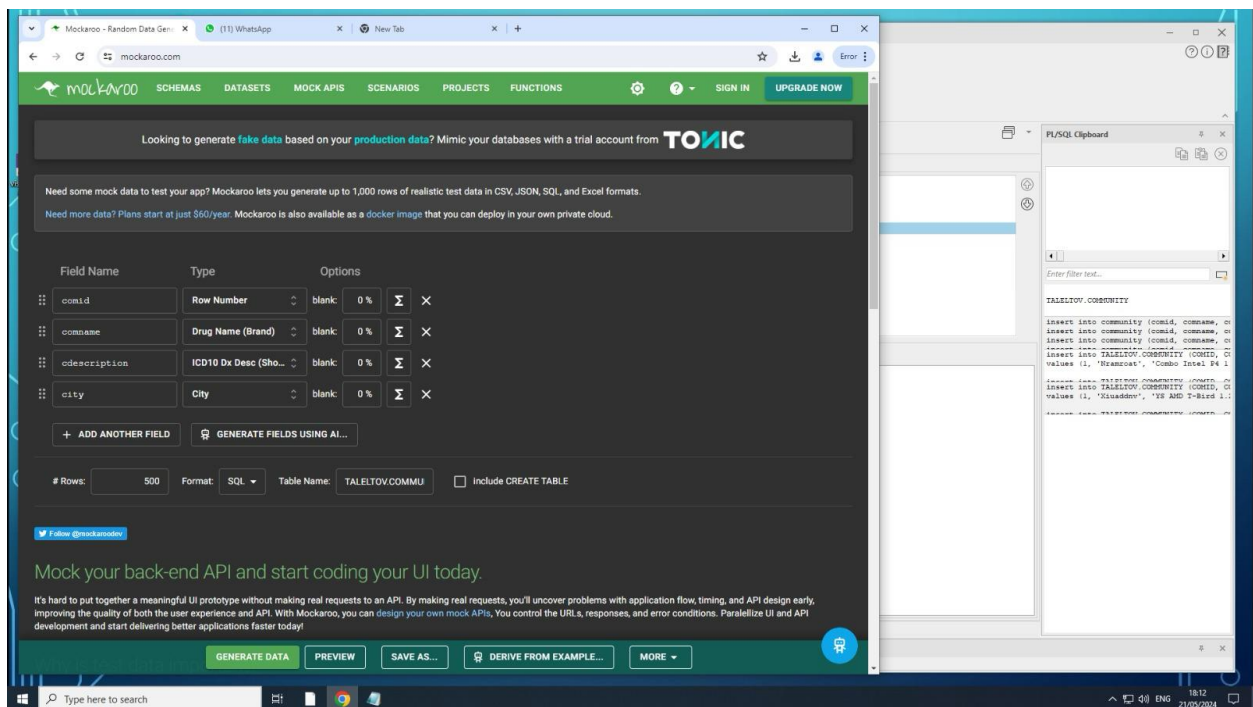
עבור members:

```
CREATE TABLE Members
(
    ComId INT NOT NULL,
    MainTel VARCHAR2(10) NOT NULL,
    PRIMARY KEY (ComId, MainTel),
    FOREIGN KEY (ComId) REFERENCES Community(ComId),
    FOREIGN KEY (MainTel) REFERENCES CommunityMember(MainTel)
);
```

	Name	Type	Nullable	Default	Comments	its
►	COMID	INTEGER				
2	MAINTEL	VARCHAR2(10)				

שיטות להכנסת נתונים:

:mackeroo



insert into TALELTOV.COMMUNITY (comid, comname, cdescription, city) values (1, 'SUPER ORIGIN COTTON FINISH FOUNDATION', 'Unsp trochan fx unsp femr, 7thM', 'Bolian');

insert into TALELTOV.COMMUNITY (comid, comname, cdescription, city) values (2, 'Prometh with Codeine', 'Laceration of other part of colon, subsequent encounter', 'Lykeio');

insert into TALELTOV.COMMUNITY (comid, comname, cdescription, city) values (3, 'Sulindac', 'Toxic effect of other metals, undetermined, init encntr', 'Aṭ Ṭawāḥīn');

insert into TALELTOV.COMMUNITY (comid, comname, cdescription, city) values (4, 'Tussin Original', 'Female genital prolapse', 'Changsheng');

insert into TALELTOV.COMMUNITY (comid, comname, cdescription, city) values (5, 'MINERAL FOUNDATION', 'Nondisp fx of prox phalanx of I little fngr, init for opn fx', 'Dodu Dua');

:generatedata.com

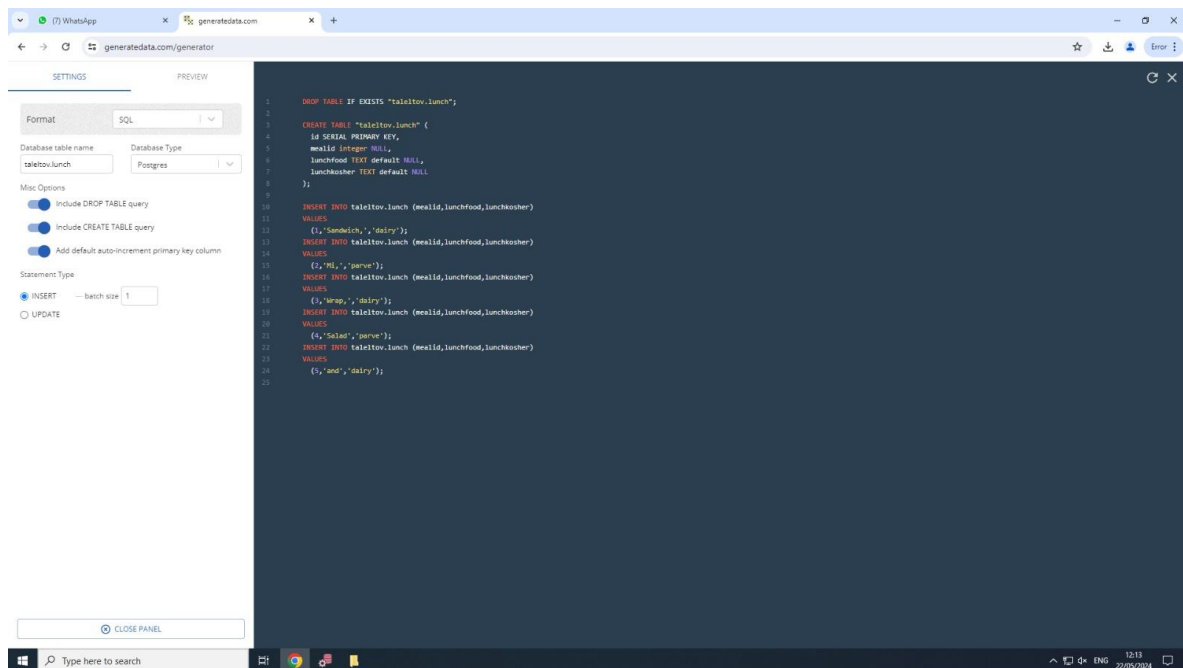
The screenshot shows the generatedata.com website interface. At the top, there's a navigation bar with links for Home, News, Generator, Register, and Login. The main section is titled 'New Data Set' and contains a table with columns: Data Type, Column Name, Examples, and Options. The table has four rows:

	Data Type	Column Name	Examples	Options
#1	Auto-increment	mealid	Select...	Start at: 1, Increment: 1
#2	Random Number of W...	lunchfood	No examples available	Generate 1 to 1 WORDS
#3	Random Number of W...	lunchkoshier	No examples available	Generate 1 to 1 WORDS
#4	Select...			

Below the table, there's an 'Add 1 ROW' button. At the bottom of the page, there's a 'POSTGRES' section showing a generated SQL script:

```
1 DROP TABLE IF EXISTS "taletov.lunch";
2
3 CREATE TABLE "taletov.lunch" (
4   id SERIAL PRIMARY KEY,
5   mealid integer NULL,
6   lunchfood TEXT default NULL,
7   lunchkoshier TEXT default NULL
8 );
9
10 INSERT INTO taletov.lunch (mealid,lunchfood,lunchkoshier)
11 VALUES
12 (1,'Sandwich','dairy');
13 INSERT INTO taletov.lunch (mealid,lunchfood,lunchkoshier)
14 VALUES
15 (2,'Me','parve');
16 INSERT INTO taletov.lunch (mealid,lunchfood,lunchkoshier)
17 VALUES
18 (3,'Wpaw','dairy');
19
```

The bottom of the page features a footer with a search bar, social media icons, and a date/time stamp: 12:31 22/05/2024.



INSERT INTO taletov.lunch (mealid,lunchfood,lunchkosher)

VALUES

(1,'Wrap','parve');

INSERT INTO taletov.lunch (mealid,lunchfood,lunchkosher)

VALUES

(2,'Wrap','parve');

INSERT INTO taletov.lunch (mealid,lunchfood,lunchkosher)

VALUES

(3,'BLT','parve');

INSERT INTO taletov.lunch (mealid,lunchfood,lunchkosher)

VALUES

(4,'Sausage','dairy');

INSERT INTO taletov.lunch (mealid,lunchfood,lunchkosher)

VALUES

(5,'Grilled','parve');

:data genertor tools

SelectTables.sql DropTable.sql TALELTOV.COMMUNITY@LABDBWIN birthDatagenerator.gd

MEAL

Owner	Table	Number of records
TALELTOV	MEAL	500

Name	Type	Size	Data	Master
MEALID	NUMBER		Sequence(1, [Inc], [WithinParent])	...
MEALDATE	DATE		List(select bdate from birth)	...
MAINTEL	VARCHAR2	10	List(select maintel from communitymember ...)	...
BNUM	NUMBER		List(select bnum from birth)	...
*				...

```

insert into TALELTOV.DINNER (MEALID, DINNERFOOD, DINNERKOSHER)
values (1, 'Cranberry', 'meat');

insert into TALELTOV.DINNER (MEALID, DINNERFOOD, DINNERKOSHER)
values (2, 'Sandwich,', 'meat');

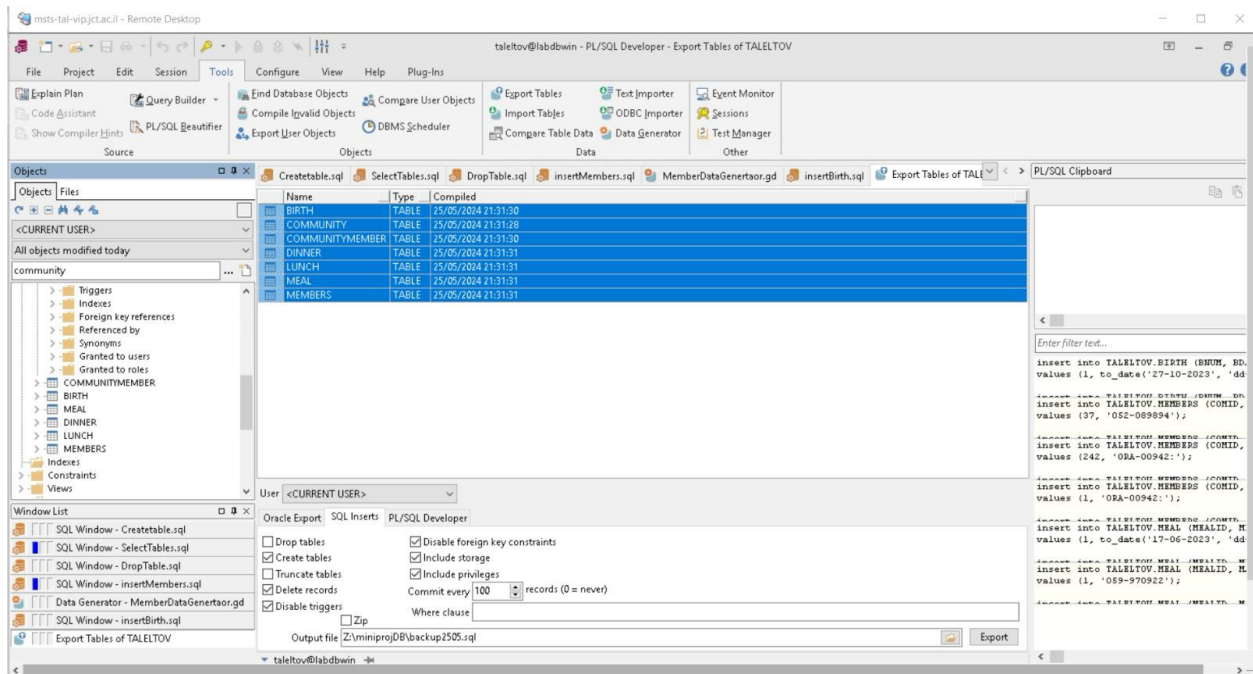
insert into TALELTOV.DINNER (MEALID, DINNERFOOD, DINNERKOSHER)
values (3, 'Roast', 'parve');

insert into TALELTOV.DINNER (MEALID, DINNERFOOD, DINNERKOSHER)
values (4, 'Mozzarella', 'meat');

insert into TALELTOV.DINNER (MEALID, DINNERFOOD, DINNERKOSHER)
values (5, 'Wrap,', 'parve');

```

גיבוי:



שלב ב'

:queries

:select

1. הצגה של כל הארוחות שהיו לפני 60 יום(מהתאריך הנוכחי) עוזר לראות מה היו כל הארוחות ביום ספציפי.
ההרצה:

```
--select all meals 60 days ago
select mealid
from meal
where TRUNC(mealdate)=(TRUNC(SYSDATE)-60)
```

התוצאה:

	MEALID	
▶ 1	13	
2	245	
3	449	

2. הצגת כל הארוחות שפרווה (גם צהריים וגם ערב)

ההרצה:

```
--select all meals that are parve
select mealid,food
from (
(select mealid as mealid, lunchfood as food
from lunch
where lunchkosher='parve')
union
(select mealid as mealid, dinnerfood as food
from dinner
where dinnerkosher='parve')
);
```

התוצאה:

	MEALID	FOOD	
▶ 1	1	Wrap,	...
2	2	Wrap,	...
3	3	BLT	...
4	3	Roast	...
5	5	Grilled	...

3. הצגה של כל הקהילות עם לפחות 5 חברי קהילה

ההרצה:

```
--select all community with at least 5 community memebers

select distinct *
from community
where comid in(
select comid
from communitymember
group by comid
having count(*)>=4);
```

התוצאה:

	COMID ▲	COMNAME		COMAMOUNT		CITY	
14	325	Bumetanide	...	5	...	Stegna	...
13	315	Hyoscyamine Sulfate	...	4	...	Shawan	...
12	304	Sucralfate	...	4	...	Tokonou	...
11	297	Chorionic Gonadotropin	...	4	...	Kushnar?nkovo	...
10	290	Carbamazepine	...	4	...	Stockholm	...

4. הצגה כל הלידות שאין להם עדיין ארוחות, יכול לעזור לניהול ארוחות

ההרצה:

```
--select births with no meals yet

select distinct *
from birth
where bnum in(
select bnum
from meal
group by bnum
having count(*)<1);
```

התוצאה:

	BNUM	BDATE	MEALCOUNT	MAINTEL	
--	------	-------	-----------	---------	--

:delete

1. מחיקת 5 הארוחות ערב שהכי פחות הכינו, כלומר שאהבו אותם הכי פחות.

ההרצה:

```
--delete
--delete 5 dinners people didn't like(showed up least amount of times)
delete
from dinner
where dinnerfood in(
  select dinnerfood
  from(
    select dinnerfood
    from(
      select dinnerfood, count(*) as amount
      from dinner
      group by dinnerfood
      order by count(*) asc
    )
    where ROWNUM<5
  )
);
--delete 5 earliest dinners
```

בסיס הנתונים לפני:

	DINNERFOOD	AMOUNT
1	Southwest	1
2	Po'	1
3	(Bacon,	1
4	Cream	1
5	Korean	1
6	Pork	1
7	Shawarma	1
8	Ranch	1
9	Sprout	1
10	Boy,	1
11	Peanut	1
12	Curried	1
13	Tikka	1
14	Rye,	1
15	with	1
16	Dip	1

בסיס הנתונים אחרי:

	DINNERFOOD	AMOUNT
1	Shawarma	1
2	Ranch	1
3	Sprout	1
4	Pulled	1
5	Rye,	1
6	Tikka	1
7	with	1
8	Korean	1
9	Pork	1
10	Southwest	1
11	Hoagie,	2
12	Parmesan	2

2. מחיקת 5 הלידות הכי ישנות, כלומר שקרו לפני הכי הרבה זמן, והכי פחות רלוונטיות לנו.

ההרצה:

```
--delete 5 earliest births
delete
from birth
where bdate in(
    select bdate
    from(
        select bdate
        from(
            select bdate
            from birth

            order by bdate asc
        )
        where ROWNUM<6
    )
);
```

בסיס הנתונים לפני:

	BDATE	
1	01/05/2023	...
2	02/05/2023	...
3	02/05/2023	...
4	03/05/2023	...
5	04/05/2023	...

בסיס הנתונים אחרי:

	BNUM	BDATE	MEALCOUNT	MAINTEL
12	483	05/05/2023	7	059-604637
14	485	17/05/2023	6	058-109987
3	474	05/06/2023	9	052-956350
7	478	09/06/2023	6	051-469545
15	486	01/07/2023	4	056-775221
4	475	02/10/2023	8	052-221144
2	473	14/11/2023	7	058-227137
10	481	30/11/2023	6	057-212124
1	472	11/12/2023	5	058-375221
9	480	27/12/2023	8	051-259952
8	479	14/01/2024	6	051-855531
13	484	19/02/2024	5	054-157203
17	488	22/02/2024	7	059-585029
6	477	25/02/2024	2	054-586120
16	487	10/03/2024	1	059-573455

:update

1. עדכון כמות הארוחות עבור יולדת להיות אפס אם הלידה לא מופיעה בטבלאות של הארוחות.

ההרצה:

```
--update mealcount to be zero if birth has no meals
update birth
set mealcount=0
where bnum in(
    select distinct bnum
    from birth
    where bnum in(
        select bnum
        from meal
        group by bnum
        having count(*)<1)
);
```

בסיס הנתונים לפני:

	BNUM	BDATE	MEALCOUNT	MAINTEL
320	291	11/04/2024	1	053-911709
360	331	01/02/2024	1	052-876843
391	362	12/11/2023	1	053-824186
144	85	08/04/2024	1	053-690487
364	335	22/02/2024	1	053-559323
322	293	21/07/2023	1	056-637999

בסיס הנתונים אחרי: (לא קיים אצלנו כרגע לידה בלי ארוחה ולכן זה ללא שינוי)

	BNUM	BDATE	MEALCOUNT	MAINTEL
320	291	11/04/2024	1	053-911709
360	331	01/02/2024	1	052-876843
391	362	12/11/2023	1	053-824186
144	85	08/04/2024	1	053-690487
364	335	22/02/2024	1	053-559323
322	293	21/07/2023	1	056-637999
218	189	05/03/2024	1	058-527606
386	357	13/09/2023	1	053-051033

2. עדכון תיאור קהילה להיות מספר חברי הקהילה שלה.

ההרצה:

```
--update cdescription of community to be the amount of members it has
update community
set cdescription=(
    select count(*)
    from communitymember
    where communitymember.comid=community.comid)
```

בסיס הנתונים לפני:

	COMID	COMNAME	CDESCRIPTION	CITY
1	49	Leucovorin Calcium	Transient retinal artery occlusion	Yichun
2	50	Clonidine Hydrochloride	Disp fx of proximal phalanx of right lesser toe(s), sequela	Newmarket
3	51	Concealing Spot Treatment	Striking against wall of swimming pool causing drown, init	Vacoas
4	52	Clobetasol Propionate	Burn 2nd deg mul sites of right lower limb, ex ank/ft, subs	Otaslavice
5	53	Fetzima	Chronic myringitis, right ear	Bayan-Ovoo
6	54	Amlodipine Besylate	Unspecified injury of body of pancreas, sequela	Alta
7	55	DIANTHUS LANCEOLATA POLLEN	Absorption of lip and oral cavity	Melituboo

בסיס הנתונים אחרי:

	COMID	COMNAME	CDESCRIPTION	CITY
1	49	Leucovorin Calcium	1	Yichun
2	50	Clonidine Hydrochloride	2	Newmarket
3	51	Concealing Spot Treatment	0	Vacoas
4	52	Clobetasol Propionate	1	Otaslavice
5	53	Fetzima	3	Bayan-Ovoo

:alter tables

1. ערך ברירת מחדל עבור מספר בית עבור חבר קהילה יהיה 0

```
-- default homenumber for community is 0
alter table COMMUNITYMEMBER modify homenumber default 0;

insert into TALELTOV.COMMUNITYMEMBER (MAINTEL, STREET, CITY, COMNAME, USERNAME, PASSWORD, COMID)
values ('051-970922', '27 Walsh Street', 'Norton', 'Lucky Instant Hand S', 'Jfxxdgr', '909814754', 214);
select * from communitymember
where comid=214
```

בדקנו ע"י הכנסת חבר קהילה בלי מספר בית, ואכן ניתן לראות שמספר הבית שלו מוגדר להיות 0:

	MAINTEL	STREET	CITY	HOMENUMBER	COMNAME	USERNAME	PASSWORD	COMID
1	059-970922	27 Walsh Street	Norton	127	Lucky Instant Hand S	Jfxxdgr	909814754	214
2	051-244252	712 Netanya Blvd	Lyon	182	Trimethoprim	Htljmir	854655334	214
3	051-970922	27 Walsh Street	Norton	0	Lucky Instant Hand S	Jfxxdgr	909814754	214

2. שינוי שם העמודה cdescription בטבלת community להיות comamount במידה ורצינו שתכיל את מספר האנשים בקהילה.

```
-- change column name of cdescription to comAmount
alter table COMMUNITY rename column cdescription to comAmount;
select * from community
```

ואכן ניתן לראות ששם העמודה השתנה בהתאם:

	COMID	COMNAME	COMAMOUNT	CITY
1	49	Leucovorin Calcium	Transient retinal artery occlusion	Yichun
2	50	Clonidine Hydrochloride	Disp fx of proximal phalanx of right lesser toe(s), sequela	Newmarket
3	51	Concealing Spot Treatment	Striking against wall of swimming pool causing drown, init	Vacoas
4	52	Clobetasol Propionate	Burn 2nd deg mul sites of right lower limb, ex ank/ft, subs	Otaslavice
5	53	Fetzima	Chronic myringitis, right ear	Bayan-Ovoo

3. ערך ברירת מחדל לכמות ארוחות ללידה 0- כך שאם מכניסים לידה חדשה, הגיוני שאין לה עדיין ארוחות.

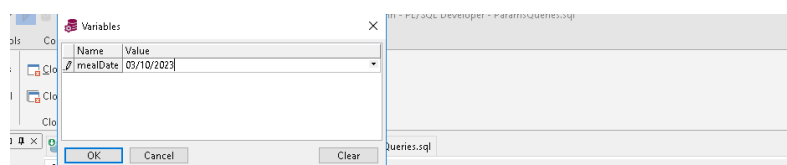
```
-- default meal count for birth is 0
alter table BIRTH modify mealcount default 0;

insert into TALELTOV.BIRTH (BNUM, BDATE, MAINTEL)
values (501, to_date('06-05-2023', 'dd-mm-yyyy'), '055-940900');
select *
from birth
where bnum=501
```

הכנסו לידה חדשה בלי להגדיר לה כמות ארוחות ואכן ראינו שמכנס 0 בשדה זה.

	BNUM	BDATE	MEALCOUNT	MAINTEL
1	501	06/05/2023	0	055-940900

Params Queries:



1. הצגת כל הארוחות (צהריים וערב) על פי תאריך מסוים שהמשתמש בוחר.
ההרצה:

התוצאה:

	MEALID	FOOD	MEALDATE
1	2	Sandwich,	03/10/2023
2	2	Wrap,	03/10/2023
3	131	Brie	03/10/2023
4	131	Tuna	03/10/2023
5	219	Sandwich,	03/10/2023
6	489	Sandwich,	03/10/2023
7	298	Sandwich,	03/10/2023
8	457	Turkey	03/10/2023

2. הצגת פרטי כל חברי הקהילות מעיר מסוימת שהמשתמש בוחר.
ההרצה:

SQL Output Statistics

```
-- הענת כל החברי קהילות לפי עיר מסוימת
SELECT username, comname, street, city
FROM communitymember
WHERE city = <name=city type=string hint="Enter the city name" default="Kindersley">
ORDER BY username;
```

Variables

Name	Value
city	Kindersley

OK Cancel Clear

Enter the city name

התוצאה:

	USERNAME	COMNAME	STREET	CITY
1	Mkumjoe	Zolpidem Tartrate	67 Michendorf Road	Kindersley
2	Pakkpty	Burn Mist	28 Sweeney Ave	Kindersley
3	Pddcona	Methscopolamine Brom	9 Damon Drive	Kindersley

3. הצגת כל הלידות שהיו בפרק זמן מסוים שהשתמש בוחר (המשתמש מכניס תאריך ראשון ותאריך שני והלידות שיוצגו יהיו בטווח שבניהם).

ההרצה:

```
-- הצגת כל הלידות שהיו בפרק זמן מסוים
SELECT b.Bnum, b.Bdate, b.MealCount
FROM Birth b
WHERE b.Bdate BETWEEN TO_DATE(<name=datefrom type=string hint="Start date" required=true>, 'DD/MM/YYYY')
AND TO_DATE(<name=dateto type=string hint="End date" required=true>, 'DD/MM/YYYY')
ORDER BY b.Bdate DESC;
```

Variables

Name	Value
datefrom	01.09.2023
dateto	01.11.2023

OK Cancel Clear

Start date

התוצאה:

	BNUM	BDATE	MEALCOUNT
1	442	01/11/2023	1
2	494	01/11/2023	5
3	365	01/11/2023	7
4	39	30/10/2023	3
5	495	29/10/2023	8
6	384	29/10/2023	3
7	251	29/10/2023	5
8	156	27/10/2023	1
9	57	27/10/2023	8
10	209	27/10/2023	2
11	47	26/10/2023	4
12	58	25/10/2023	9
13	214	24/10/2023	9
14	198	20/10/2023	5
15	8	20/10/2023	8
16	127	18/10/2023	7
17	363	18/10/2023	7
18	4	17/10/2023	4
19	396	16/10/2023	1

4. הצגת כל חברי הקהילה לפי שם קהילה מסוים שהשתמש בוחר.

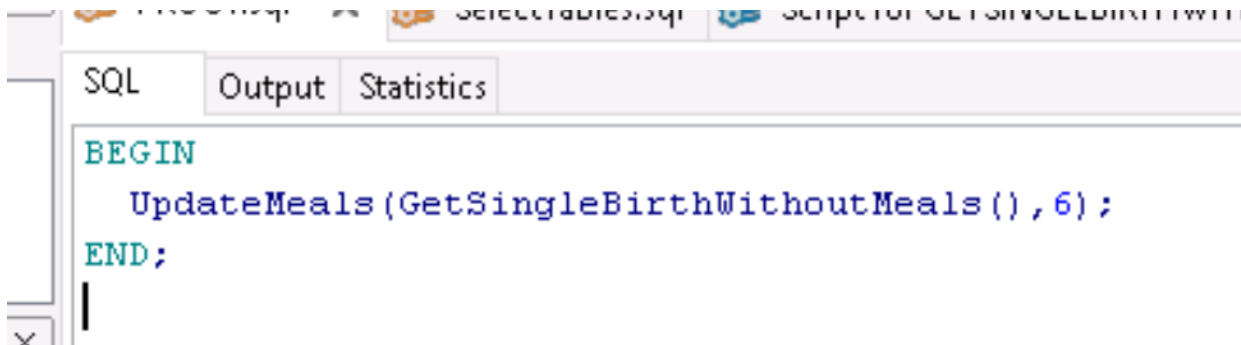
ההרצה:

התוצאה:

	MAINTEL	STREET	CITY	HOMENUMBER	COMNAME	USERNAME	PASSWORD	COMNAME	CDESCRIPTION
1	059-562801	12nd Street	Mengjia	162	Flawless Finish Dual	Rqglgqd	839359447	Metoprolol Succinate	Stenosis due to nen
2	059-610414	21 Streep Street	Naxos	34	PHOSLYRA	TbdvIhy	465750660	Metoprolol Succinate	Stenosis due to nen
3	056-066731	40 Schiavelli Drive	Al ??qir	174	COTTONTAILS	Xikfzkl	796639658	Metoprolol Succinate	Stenosis due to nen

תוכנית:

תוכנית ראשונה:



```
SQL Output Statistics
BEGIN
  UpdateMeals (GetSingleBirthWithoutMeals (), 6);
END;
```

התוכנית קוראת לפרוצדורה update meals - מוסיפה מספר ארוחות (לפי הפרמטר שקיבלנו) לטבלת meal, ארוחה לכל יום החל מעכשיו ועד למספר הימים של הפרמטר, ומוסיפה את הארוחה המתאימה לטבלת ארוחות הערב. היא מקבלת את מספר הלידה מהפונקציה שמחזירה את מספר הלידה הראשון שלא קיימות לו כרגע ארוחות.

הפונקציה-

```
CREATE OR REPLACE FUNCTION GetSingleBirthWithoutMeals
RETURN birth.bnum%TYPE
IS
  v_bnum birth.bnum%TYPE;
BEGIN
  SELECT b.bnum
  INTO v_bnum
  FROM birth b
  WHERE NOT EXISTS (SELECT 1 FROM dinner d WHERE d.bnum = b.bnum)
  AND NOT EXISTS (SELECT 1 FROM lunch l WHERE l.bnum = b.bnum)
  FETCH FIRST 1 ROW ONLY;

  RETURN v_bnum;
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    RETURN NULL;
END;
```

הפרוצדורה-

הפרוצדורה ראשית מגדירה "משתנה" עם הלידה הרצויה, אחר כך היא רצה בלולאה מספר הפעמים שקיבלה בתור פרמטר ומוסיפה ארוחות לימים הקרובים (גם בטבלת הארוחה וגם בטבלת הארוחות ערב).

```
CREATE OR REPLACE PROCEDURE UpdateMeals(bnum_sent IN INTEGER, AMOUNT IN INTEGER)
IS
  -- Define cursor to select birth where bnum is what was sent to the procedure
  CURSOR bnum_cur IS
    SELECT bnum, maintel, bdate
    FROM birth
    WHERE bnum = bnum_sent;
  -- Declare variables to hold the values fetched by the cursor
  v_bnum birth.bnum%TYPE;
  v_maintel birth.maintel%TYPE;
  v_bdate birth.bdate%TYPE;
  -- Variable to hold the new mealid
  new_mealid meal.mealid%TYPE;
```

```

-- Custom exception for handling no data found
ex_no_data_found EXCEPTION;
BEGIN
-- Enable DBMS_OUTPUT
DBMS_OUTPUT.ENABLE;
-- Open the cursor and fetch data
OPEN bnum_cur;
FETCH bnum_cur INTO v_bnum, v_maintel, v_bdate;
-- Raise an exception if no rows are found
IF bnum_cur%NOTFOUND THEN
    RAISE ex_no_data_found;
END IF;
-- Close the cursor
CLOSE bnum_cur;
-- Loop to insert meals for the specified number of days
FOR i IN 1..AMOUNT LOOP
    BEGIN
        -- Generate new mealid
        SELECT NVL(MAX(mealid), 0) + 1 INTO new_mealid FROM meal;

        -- Insert into meal table
        INSERT INTO meal(mealid, mealdate, maintel, bnum)
        VALUES (
            new_mealid,
            SYSDATE + i,
            (SELECT maintel FROM communitymember ORDER BY dbms_random.value FETCH FIRST 1 ROW ONLY),
            v_bnum
        );
        -- Insert into dinner table using the same mealid
        INSERT INTO dinner(mealid, dinnerfood, dinnerkosher)
        VALUES (
            new_mealid,
            (SELECT dinnerfood FROM dinner ORDER BY dbms_random.value FETCH FIRST 1 ROW ONLY),
            (SELECT dinnerkosher FROM dinner ORDER BY dbms_random.value FETCH FIRST 1 ROW ONLY)
        );
        -- Output a message indicating success
        DBMS_OUTPUT.PUT_LINE('Inserted meal for date: ' || TO_CHAR(SYSDATE + i, 'YYYY-MM-DD') || ' with mealid: ' || new_mealid);
    EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Error inserting: ' || SQLERRM);
    END;
END LOOP;
-- Commit the transaction
COMMIT;
EXCEPTION
    WHEN ex_no_data_found THEN
        DBMS_OUTPUT.PUT_LINE('No record found for birth number: ' || bnum_sent);
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('An unexpected error occurred: ' || SQLERRM);
END;

```

/

כשמרצים רק את הפונקציה

test script DBMS Output Statistics Profiler Trace

```

1 begin
2   -- Call the function
3   :result := getsinglebirthwithoutmeals;
4 end;

```

Variable	Type	Value
result	Float	345

1:1 taleltov@labdbwin Executed in 0.006 seconds

ובאמת לא קיימים ארוחות ללידה זו

SQL Output Statistics

```

select *
from meal
where bnum=345;

```

MEALID	MEALDATE	MAINTEL	BNUM
--------	----------	---------	------

15:1 taleltov@labdbwin 0 rows selected in 0.013 seconds

ולאחר שמריצים את התוכנית ניתן לראות שקיימים 6 ארוחות עבור 6 הימים הקרובים

SQL Output Statistics

```

select *
from meal
where bnum=345;

```

	MEALID	MEALDATE	MAINTEL	BNUM
1	539	01/07/2024 01:01:20	053-867571	345
2	540	02/07/2024 01:01:20	051-272916	345
3	541	03/07/2024 01:01:20	057-334309	345
4	542	04/07/2024 01:01:20	053-939149	345
5	543	05/07/2024 01:01:20	059-533439	345
6	544	06/07/2024 01:01:20	058-852218	345

14:3 taleltov@labdbwin 6 rows selected in 0.020 seconds

כשנריץ בפעם הבאה את הפרוצדורה נראה שקיבלנו מספר לידה אחר (כי ללידה הקודמת כבר יש ארוחות)

Test script DBMS Output Statistics Profiler Trace

```

1 begin
2   -- Call the function
3   :result := getsinglebirthwithoutmeals;
4 end;

```

	Variable	Type	Value
▶	result	Float	504
*			

1:1 taleltov@labdbwin Executed in 0.003 seconds

תכנית שנייה:

תיאור מילולי:

התכנית הנ"ל נועדה להדפיס מידע המתמקד בשתי קהילות מהקהילות במאגר. מטרתה לבצע העברה של חברי קהילה מקהילה אחת (קהילת ה'מקור') לקהילה אחרת (קהילת ה'יעד'), ולהדפיס שוב את המידע על הקהילות לאחר ההעברה. המידע המודפס כולל את מספר חברי הקהילה וממוצע הארוחות לחבר קהילה בכל קהילה. שימו לב- לאחר העברת חברי קהילה מקהילת המקור לקהילת היעד, קהילת המקור נותרת ריקה מחברים, ולכן בהכרח תמיד לאחר ההעברה מספר חברי הקהילה וממוצע ארוחות לחבר בקהילת המקור יהיו אפס.

התכנית כוללת פונקציות ופרוצדורות המיועדות לחישוב והדפסה של מידע זה (כמו

הפונקציה `GetCommunityMemberCount`

המקבלת את מספר הקהילה ומחזירה את מספר החברים בה, וכמו הפרוצדורה

`PrintCommunityDetails`

המדפיסה את הפרטים על הקהילות לפני ואחרי ההעברה), תוך טיפול בחריגות שעלולות להתרחש במהלך הביצוע.

את החישוב עצמו של ממוצע ארוחות לאדם בקהילה לכל אחת מן הקהילות התכנית עושה באמצעות קריאה לפונקציה

`ReturnAverageMealsPerCommunity`.

ואילו את ההעברה של כל חברי הקהילה מקהילת המקור ליעד התכנית מבצעת באמצעות קריאה לפרוצדורה

`TransferCommunityMembers`

קוד:

DECLARE

```
fromCommunityId Community.ComId%TYPE := 1; -- ראשונה קהילה מספר-1
toCommunityId Community.ComId%TYPE := 2; -- שנייה קהילה מספר-2
avgMealsBeforeFrom NUMBER;
avgMealsBeforeTo NUMBER;
avgMealsAfterFrom NUMBER;
avgMealsAfterTo NUMBER;
memberCountBeforeFrom NUMBER;
memberCountBeforeTo NUMBER;
memberCountAfterFrom NUMBER;
memberCountAfterTo NUMBER;

-- הקהילה חברי מספר להחזרת פונקציה
FUNCTION GetCommunityMemberCount(communityId IN Community.ComId%TYPE)
RETURN NUMBER IS
    memberCount NUMBER;
BEGIN
    SELECT COUNT(*)
    INTO memberCount
    FROM CommunityMember
    WHERE ComId = communityId;
    RETURN memberCount;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RETURN 0;
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Error occurred: ' || TO_CHAR(SQLCODE) || '
- ' || SUBSTR(SQLERRM, 1, 200));
        RETURN 0;
END GetCommunityMemberCount;
```

```

-- הקהילה פרטי להדפסת פרוצדורה
PROCEDURE PrintCommunityDetails (communityId IN Community.ComId%TYPE,
beforeTransfer BOOLEAN) IS
    memberCount NUMBER;
    avgMeals NUMBER;
    beforeAfter VARCHAR2 (20);
BEGIN
    memberCount := GetCommunityMemberCount (communityId);
    avgMeals := ReturnAverageMealsPerCommunity (communityId);
    beforeAfter := CASE WHEN beforeTransfer THEN 'before transfer' ELSE
'after transfer' END;
    DBMS_OUTPUT.PUT_LINE ('Community ' || communityId || ' ' || beforeAfter
|| ' - ' );
    DBMS_OUTPUT.PUT_LINE ('Number of members: ' || memberCount);
    DBMS_OUTPUT.PUT_LINE ('Average meals per member: ' || avgMeals);
    DBMS_OUTPUT.PUT_LINE ('');
END PrintCommunityDetails;

BEGIN
-- ההעברה לפני הקהילות פרטי הדפסת
PrintCommunityDetails (fromCommunityId, TRUE);
PrintCommunityDetails (toCommunityId, TRUE);

-- הקהילה חברי העברת
TransferCommunityMembers (fromCommunityId, toCommunityId);

-- ההעברה אחרי הקהילות פרטי הדפסת
PrintCommunityDetails (fromCommunityId, FALSE);
PrintCommunityDetails (toCommunityId, FALSE);

END;

```

צילומי מסך נחוצים:

```

SELECT cm.MainTel, cm.Username, cm.ComId, COUNT(m.MealID) AS NumberOfMeals
FROM CommunityMember cm
LEFT JOIN Meal m ON cm.MainTel = m.MainTel
WHERE cm.ComId = 170 -- מספר הקהילה שנוצרה לבדיקה
GROUP BY cm.MainTel, cm.Username, cm.ComId
ORDER BY cm.MainTel;

```

	MAINTEL	USERNAME	COMID	NUMBEROFMEALS
1	054-875872	Yoraszv	170	1
2	057-613269	Tynicdz	170	0
3	057-647270	Fdgpdqp	170	1

```
SELECT cm.MainTel, cm.Username, cm.ComId, COUNT(m.MealID) AS NumberOfMeals
FROM CommunityMember cm
LEFT JOIN Meal m ON cm.MainTel = m.MainTel
WHERE cm.ComId = 180 -- מספר הקהילה שגרעה לברוק
GROUP BY cm.MainTel, cm.Username, cm.ComId
ORDER BY cm.MainTel;
```

Select community	Select communitymember	Select birth	Select meal	Select lunch	Select dinner	Select members	Select
MAINTEL	USERNAME	COMID	NUMBEROFMEALS				
1 053-990149	Jwkytjz	180	0				
2 055-442153	Licwgss	180	1				
3 056-132176	hwjbhnm	180	0				

בחרנו בשתי קהילות רנדומליות מספר 170 ו 180.
 ניתן לראות כי לפני הרצת התכנית, בקהילה מס 170 יש 3 חברים ועבור 2 מתוכם יש ארוחה אחת לכל אחד.
 בדומה לכך בקהילה מס 180 יש גם 3 חברים אלא שרק לאחד מהם יש ארוחה אחת בלבד.
 כלומר אם נחשב לבד נגיע לממוצע ארוחה לאדם של 0.66 בקהילה מס 170 ו 0.33 בקהילה מס 180.
 כעת נריץ את התכנית כאשר קהילת המקור (שתרוקן מחברים) היא מס 170 וקהילת היעד היא מס 180.

```
Test script DBMS Output Statistics Profiler Trace
1 DECLARE
2   fromCommunityId Community.ComId%TYPE := 170; -- מספר קהילה ראשונה
3   toCommunityId Community.ComId%TYPE := 180; -- מספר קהילה שנייה
4   avgMealsBeforeFrom NUMBER;
5   avgMealsBeforeTo NUMBER;
6   avgMealsAfterFrom NUMBER;
7   avgMealsAfterTo NUMBER;
8   memberCountBeforeFrom NUMBER;
9   memberCountBeforeTo NUMBER;
10  memberCountAfterFrom NUMBER;
11  memberCountAfterTo NUMBER;
12
13  -- פונקציה להחזרת מספר חברי הקהילה
14  FUNCTION GetCommunityMemberCount (communityId IN Community.ComId%TYPE) RETURN NUMBER IS
15    memberCount NUMBER;
16  BEGIN
17    SELECT COUNT(*)
18    INTO memberCount
19    FROM CommunityMember
```

פלט התכנית:


```

SELECT cm.MainTel, cm.Username, cm.ComId, COUNT(m.MealID)
FROM CommunityMember cm
LEFT JOIN Meal m ON cm.MainTel = m.MainTel
WHERE cm.ComId = 170 -- מספר הקהילה שנוצרה לבדיקה
GROUP BY cm.MainTel, cm.Username, cm.ComId
ORDER BY cm.MainTel;

```

```

SELECT cm.MainTel, cm.Username, cm.ComId, COUNT(m.MealID)

```



קוד הפונקציה:

```

CREATE OR REPLACE FUNCTION ReturnAverageMealsPerCommunity(communityId IN
Community.ComId%TYPE) RETURN NUMBER IS
    totalMeals NUMBER := 0;
    memberCount NUMBER := 0;
    avgMealsPerMember NUMBER := 0;

    CURSOR memberCursor IS
        SELECT cm.MainTel
        FROM CommunityMember cm
        WHERE cm.ComId = communityId;

    memberTel VARCHAR2(10);
    mealCount NUMBER;

BEGIN

    OPEN memberCursor;

    LOOP
        FETCH memberCursor INTO memberTel;
        EXIT WHEN memberCursor%NOTFOUND;

        BEGIN
            SELECT COUNT(*)
            INTO mealCount
            FROM Meal m
            WHERE m.MainTel = memberTel;
        EXCEPTION
            WHEN NO_DATA_FOUND THEN
                mealCount := 0;
        END;

        totalMeals := totalMeals + mealCount;
        memberCount := memberCount + 1;
    END LOOP;

    CLOSE memberCursor;

```

```

    IF memberCount > 0 THEN
        avgMealsPerMember := totalMeals / memberCount;
    ELSE
        avgMealsPerMember := 0;
    END IF;

    RETURN avgMealsPerMember;

EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Error occurred: ' || TO_CHAR(SQLCODE) || ' - '
|| SUBSTR(SQLERRM, 1, 200));
        RETURN 0;
END ReturnAverageMealsPerCommunity;

```

קוד הפרוצדורה:

```

CREATE OR REPLACE PROCEDURE TransferCommunityMembers(
    fromCommunityId IN Community.ComId%TYPE,
    toCommunityId IN Community.ComId%TYPE
)
IS
    TYPE MemberRec IS RECORD (
        mainTel CommunityMember.MainTel%TYPE,
        comId CommunityMember.ComId%TYPE
    );

    CURSOR member_cursor IS
        SELECT cm.MainTel, cm.ComId
        FROM CommunityMember cm
        WHERE cm.ComId = fromCommunityId;

    member_rec MemberRec;
    err_msg VARCHAR2(32767);

BEGIN
    OPEN member_cursor;
    LOOP
        FETCH member_cursor INTO member_rec;
        EXIT WHEN member_cursor%NOTFOUND;

        UPDATE CommunityMember
        SET ComId = toCommunityId
        WHERE MainTel = member_rec.mainTel;
    END LOOP;
    CLOSE member_cursor;

    DBMS_OUTPUT.PUT_LINE('*****');
    DBMS_OUTPUT.PUT_LINE('Community members transferred successfully');
    DBMS_OUTPUT.PUT_LINE('*****');
    DBMS_OUTPUT.PUT_LINE(' ');
    COMMIT;
EXCEPTION
    WHEN OTHERS THEN

```



```
        err_msg := 'Error occurred: ' || TO_CHAR(SQLCODE) || ' - ' ||  
SUBSTR(SQLERRM, 1, 200);  
        DBMS_OUTPUT.PUT_LINE(err_msg);  
        ROLLBACK;  
END TransferCommunityMembers;
```