

דו"ח מיני"פ - חיל האוויר

מגישים: ידידיה שאולי ואורי נשר



תוכן עניינים

2	תוכן עניינים
3	תיאור הארגון במילים
4	תרשים ERD
5	DSD תרשים
6	סכמות
6	בדיקת נרמול:
7	קשרים בין ישויות
8	script של create table
10	script של drop table
11	הכנסת נתונים באמצעות Data Generator
11	Mockaroo
11	Data Generator
12	אקסל
14	גיבוי
14	גיבוי ב pl/sql בעזרת Export tables
15	גיבוי לגיט
16	שחזור בעזרת import tables
17	שלב ב
17	שאלות בחירה ללא פרמטרים
17	1. שליפת בסיסים צבאיים עם קיבולת מעל הממוצע
18	2. ספירת טיסות לפי סוג משימה
18	3. רשימת משימות מחמש השנים האחרונות לפי סוג מטוס
	תיאור: השאלתה שולפת משימות מהחמש השנים האחרונות ומציגה אותן מקובצות לפי סוג המטוס וסוג המשימה. התוצאות מסודרות לפי תאריך המשימה, סוג המשימה וסוג המטוס.
19	4. כל הטיסות מאותו סוג של הטיסה הבאה שתתרחש
20	שאלות עם פרמטרים
21	1. כל המטוסים עם סטטוס מסויים ומיקום מסויים
22	2. פרטי משימות, טיסות, מטוסים, חיילים ובסיסים עם סכומי ספקים כוללים
23	3. משימות שבוצעו על ידי חיילים של דרגה מסוימת בשנתיים האחרונות
24	4. פרטי חיילים, משימות, מטוסים ובסיסים על פי דרגת החייל, מיקום הבסיס וקיבולת מינימלית של הבסיס

שלב א'

תיאור הארגון במילים

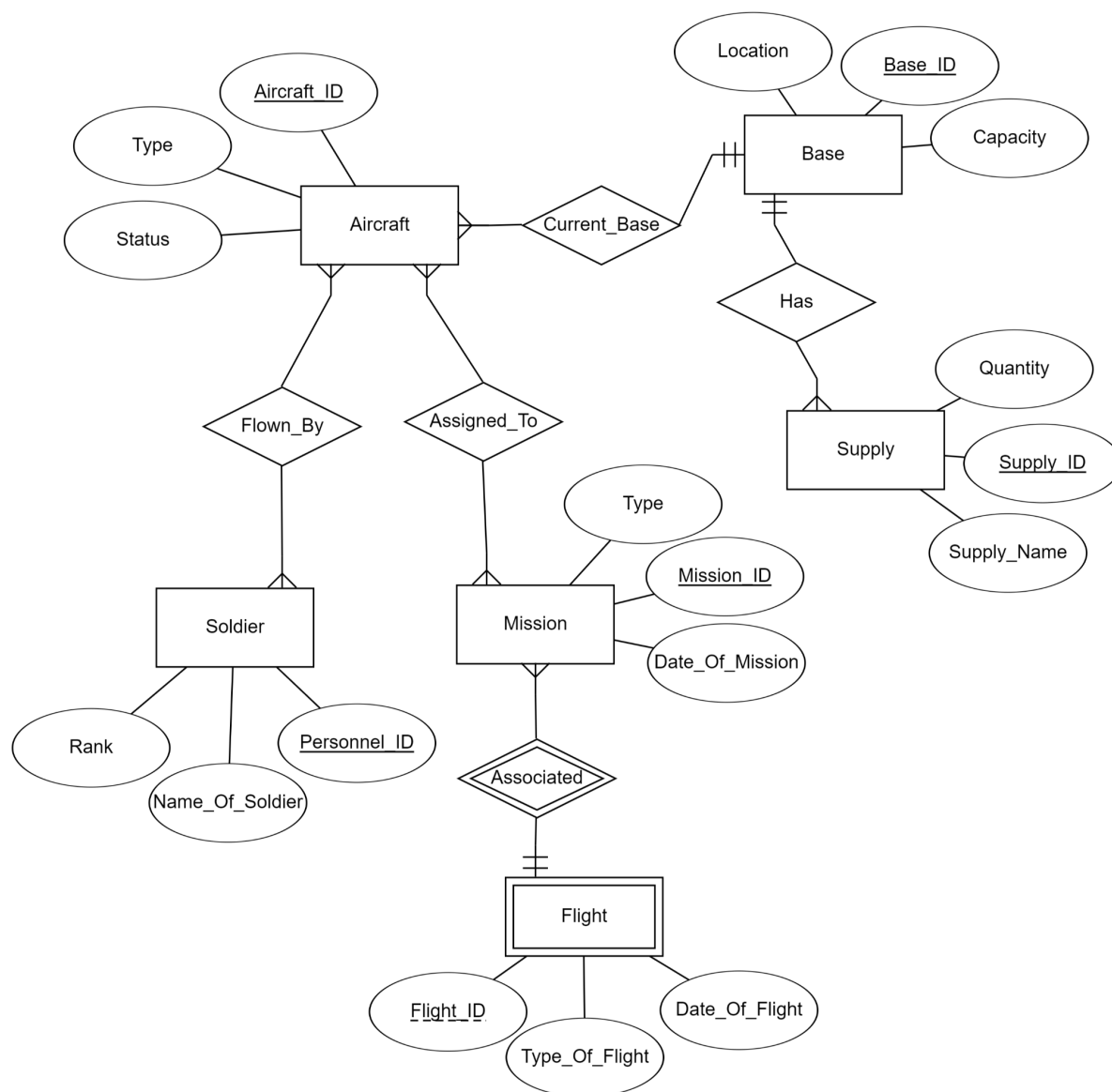
חיל האוויר של צה"ל פועל עם מערכת מסד נתונים מובנית בקפידה שנועדה לנהל את דרישותיו המבצעיות ביעילות.

מסד נתונים זה משלב ישויות חיוניות כגון מטוסים, חיילים, בסיסים, משימות, אספקה וטיסות. הוא משמש עמוד השדרה לתזמון ותיאום משימות, לוגיסטיקה ותחזוקה, ומבטיח הקצאת משאבים מיטבית. מידע מפורט על תחזוקת מטוסים, היסטוריית תפעול ומדדי ביצועים מתועד, בעוד שהגישה לנתונים רגישים מוגבלת לצוות מורשה.

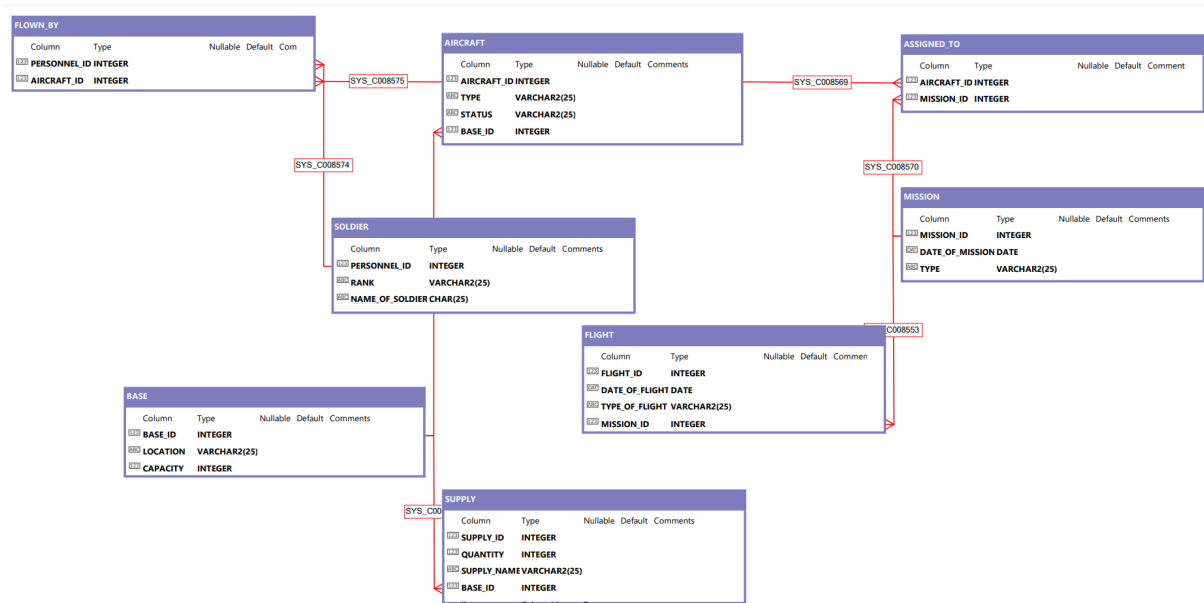
באמצעות גישה שיטתית זו, חיל האוויר משפר את מוכנותו ויעילותו המבצעית, ומקל על הצלחת המשימה.

בנוסף לשמירת נתונים כגון כמויות האספקה בכל בסיס, שמירת הבסיס הנוכחי של כל מטוס, ותיעוד הקצאת כלי הטיס למשימות השונות, המערכת גם שומרת אילו חיילים מטיסים איזה כלי טיס.

תרשים ERD



תרשים DSD



תיאור כללי של הישויות והקשרים

סכמות

Aircraft (Aircraft_ID, Type, Status, Base_ID)
Soldier (Personnel_ID, Rank, Name_Of_Soldier)
Flight (Flight_ID, Date_Of_Flight, Type_Of_Flight, Mission_ID)
Supply (Supply_ID, Quantity, Supply_Name, Base_ID)
Flown_By (Personnel_ID, Aircraft_ID)
Mission (Mission_ID, Date_Of_Mission, Type)
Base (Base_ID, Location, Capacity)
Assigned_To (Aircraft_ID, Mission_ID)

בדיקת נרמול:

- Aircraft
הסוג והסטטוס ובסיס הבית תלויים במזהה כלי הטייס ואין תלות ביניהם - מתקיים 3NF
- Soldier
הדרגה תלויה במזהה-חייל ולכן מתקיים 3NF
- Flight
התאריך והמזהה של המשימה תלויים במזהה-הטיסה ולכן מתקיים 3NF
- Supply
כמות האספקה ומזהה הבסיס תלויים במספר המזהה של האספקה ולכן מתקיים 3NF
- Flown_By
אין תכונות מלבד המפתח ולכן מתקיים 3NF
- Mission
Date תלוי במפתח, מתקיים 3NF
- Base
המיקום והקיבולת של כל בסיס תלויים במספר המזהה שלו, מתקיים 3NF
- Assigned_To
אין תכונות מלבד המפתח ולכן מתקיים 3NF

קשרים בין ישויות

- Flown_By
קשר רבים לרבים, כל חייל יכול להטיס כמה כלי טיס, וכל כלי טיס כמה חיילים יכולים להטיס אותו.
- Assigned_To
קשר רבים ליחיד (אחד לכל היותר), לכל משימה יכול להיות אפס או יותר כלי טיס שהוקצו לה, אבל לכל כלי טיס יכול להיות רק משימה אחת אליה הוא הוקצה בכל זמן נתון.
- Current_Base
קשר רבים ליחיד (בדיוק אחד), כל מטוס יכול להיות רק בבסיס אחד בכל זמן נתון (גם כשהוא יוצא למשימות עדיין יש לו בסיס אליו הוא צריך לחזור בתום המשימה) ובבסיס יכולים להיות הרבה כלי טייס שונים באותו זמן.
- Has
קשר רבים ליחיד (בדיוק אחד), כל אספקה יכולה להיות בבסיס אחד בלבד. בבסיס יכולות להיות אספקות רבות מכל מיני סוגים בבת אחת.
- Associated
לא קשר אמיתי אלא Flight היא ישות חלשה ביחס לMission כי אין טיסות שהן לא חלק ממשימה.

create table של script

```
CREATE TABLE Soldier
(
    Personnel_ID INT NOT NULL,
    Rank VARCHAR(25) NOT NULL,
    Name_Of_Soldier VARCHAR(25) NOT NULL,
    PRIMARY KEY (Personnel_ID)
);

CREATE TABLE Mission
(
    Mission_ID INT NOT NULL,
    Date_Of_Mission DATE NOT NULL,
    Type VARCHAR(25) NOT NULL,
    PRIMARY KEY (Mission_ID)
);

CREATE TABLE Base
(
    Base_ID INT NOT NULL,
    Location VARCHAR(25) NOT NULL,
    Capacity INT NOT NULL,
    PRIMARY KEY (Base_ID)
);

CREATE TABLE Flight
(
    Flight_ID INT NOT NULL,
    Mission_ID INT NOT NULL,
    Date_Of_Flight DATE NOT NULL,
    Type_Of_Flight VARCHAR(25) NOT NULL,
    PRIMARY KEY (Flight_ID, Mission_ID),
    FOREIGN KEY (Mission_ID) REFERENCES Mission(Mission_ID)
);

CREATE TABLE Supply
(
    Supply_ID INT NOT NULL,
    Quantity INT NOT NULL,
    Supply_Name VARCHAR(25) NOT NULL,
    Base_ID INT NOT NULL,
    PRIMARY KEY (Supply_ID),
```



```

FOREIGN KEY (Base_ID) REFERENCES Base(Base_ID)
);

CREATE TABLE Aircraft
(
    Aircraft_ID INT NOT NULL,
    Type VARCHAR(25) NOT NULL,
    Status VARCHAR(25) NOT NULL,
    Base_ID INT NOT NULL,
    PRIMARY KEY (Aircraft_ID),
    FOREIGN KEY (Base_ID) REFERENCES Base(Base_ID)
);

CREATE TABLE Assigned_To
(
    Aircraft_ID INT NOT NULL,
    Mission_ID INT NOT NULL,
    PRIMARY KEY (Aircraft_ID, Mission_ID),
    FOREIGN KEY (Aircraft_ID) REFERENCES Aircraft(Aircraft_ID),
    FOREIGN KEY (Mission_ID) REFERENCES Mission(Mission_ID)
);

CREATE TABLE Flown_By
(
    Personnel_ID INT NOT NULL,
    Aircraft_ID INT NOT NULL,
    PRIMARY KEY (Personnel_ID, Aircraft_ID),
    FOREIGN KEY (Personnel_ID) REFERENCES Soldier(Personnel_ID),
    FOREIGN KEY (Aircraft_ID) REFERENCES Aircraft(Aircraft_ID)
);

```

הפעלת פקודת Desc

Dialog Editor

Connected to Oracle Database 21c Express Edition Release 21.0.0.0.0
Connected as sys@xe AS SYSDBA

SQL> desc soldier

Name	Type	Nullable	Default	Comments
PERSONNEL_ID	INTEGER			
RANK	VARCHAR2(25)			
NAME_OF_SOLDIER	CHAR(25)			

SQL> desc mission

Name	Type	Nullable	Default	Comments
MISSION_ID	INTEGER			
DATE_OF_MISSION	DATE			
TYPE	VARCHAR2(25)			

SQL> desc Base

drop table של script

-- Drop the Mission table
DROP TABLE Mission;

-- Drop the Flown_By table
DROP TABLE Flown_By;

-- Drop the Base table
DROP TABLE Base;

-- Drop the Supply table
DROP TABLE Supply;

-- Drop the Flight table
DROP TABLE Flight;

-- Drop the Soldier table
DROP TABLE Soldier;

-- Drop the Aircraft table
DROP TABLE Aircraft;

-- Drop the Assigned_To table
DROP TABLE Assigned_To;

הכנסת נתונים באמצעות Data Generator

Mockaroo

ולאחר מכן פתחנו את כולם בSQL developer והרצנו את הקוד שלהם.

Aircraft
Aircraft_ID | Type | Status | Base_ID

Assigned_To
Aircraft_ID | Mission_ID

Base
Base_ID | Capacity | Location

Flight
Flight_ID | Flight_Date | Mission_ID

Flown_By
Personnel_ID | Aircraft_ID

Mission
Mission_ID | Mission_Date

Soldier
Personnel_ID | Rank

Supply
Supply_ID | Quantity | Base_ID

- Flight.sql
- Flown_By.sql
- Mission.sql
- Soldier.sql
- Base.sql
- Aircraft.sql
- Supply.sql
- Assigned_To.sql

AIRCRAFT_ID	TYPE	STATUS	BASE_ID
1	262379 Multirole	Active	251080
2	177795 Unmanned Aerial Vehicles	Inactive	423296
3	755988 Ground Attack	Inactive	924302
4	191281 Helicopters	Under Maintenance	442053
5	510192 Unmanned Aerial Vehicles	Inactive	215486
6	278348 Ground Attack	Under Maintenance	917769
7	934659 Ground Attack	Inactive	169471
8	473630 Fighter	Under Maintenance	423296
9	555924 Fighter	Active	683408
10	334561 Transport	Inactive	461130
11	832170 Unmanned Aerial Vehicles	Inactive	145151
12	896606 Multirole	Active	473743
13	292770 Ground Attack	Inactive	152260
14	945463 Multirole	Active	294580
15	468839 Transport	Under Maintenance	720297
16	383258 Multirole	Active	924302
17	678522 Fighter	Under Maintenance	220735
18	351943 Reconnaissance	Active	792636
19	136353 Reconnaissance	Inactive	515969
20	344608 Unmanned Aerial Vehicles	Inactive	673423
21	830439 Unmanned Aerial Vehicles	Under Maintenance	197463
22	375944 Reconnaissance	Inactive	198729
23	176639 Reconnaissance	Active	654758
24	755944 Transport	Under Maintenance	599239
25	572810 Multirole	Active	455571

```
insert into Base (Base_ID, Capacity, Location) values (328428, 2780, 'Ovda Airbase');
insert into Base (Base_ID, Capacity, Location) values (925940, 5532, 'Hatzerim Airbase');
insert into Base (Base_ID, Capacity, Location) values (118276, 4227, 'Hatzerim Airbase');
insert into Base (Base_ID, Capacity, Location) values (623335, 7212, 'Ovda Airbase');
insert into Base (Base_ID, Capacity, Location) values (820234, 3287, 'Ovda Airbase');
insert into Base (Base_ID, Capacity, Location) values (346528, 2030, 'Ovda Airbase');
insert into Base (Base_ID, Capacity, Location) values (797343, 5882, 'Ovda Airbase');
insert into Base (Base_ID, Capacity, Location) values (968052, 9939, 'Tel Nof Airbase');
insert into Base (Base_ID, Capacity, Location) values (232091, 4094, 'Palmachim Airbase');
insert into Base (Base_ID, Capacity, Location) values (727388, 8777, 'Ovda Airbase');
insert into Base (Base_ID, Capacity, Location) values (180217, 5253, 'Ramat David Airbase');
insert into Base (Base_ID, Capacity, Location) values (590901, 2026, 'Hatzerim Airbase');
```

Data Generator

SUPPLY				
<	Owner	Table	Number of records	
>	SYS	SUPPLY	400	
...				
	Name	Type	Size	Data
▶	SUPPLY_ID	NUMBER		Random(1000000, 9999999)
	QUANTITY	NUMBER		Random(1, 1000)
	SUPPLY_NAME	VARCHAR2	25	List('Fuel', 'Ammunition', 'Missiles', 'Bombs', 'Aircraft spare parts', 'Engines', 'Electronic components', 'Communication equipment', 'Medical supplies')
	BASE_ID	NUMBER		List(select BASE_ID from BASE)
*				









	SUPPLY_ID	QUANTITY	SUPPLY_NAME	BASE_ID
▶	8511161	946	Food rations	9568577
	5432704	317	Engines	5011067
	1939297	471	Engines	8063680
	6308945	309	Protective gear	9305786
	9735118	724	Food rations	7293252

```
insert into SYS.SUPPLY (SUPPLY_ID, QUANTITY, SUPPLY_NAME, BASE_ID)
values (2924566, 717, 'Flight manuals', 7254186);

insert into SYS.SUPPLY (SUPPLY_ID, QUANTITY, SUPPLY_NAME, BASE_ID)
values (4343398, 17, 'Food rations', 2255509);

insert into SYS.SUPPLY (SUPPLY_ID, QUANTITY, SUPPLY_NAME, BASE_ID)
values (7026982, 929, 'Logistics software', 8161420);

insert into SYS.SUPPLY (SUPPLY_ID, QUANTITY, SUPPLY_NAME, BASE_ID)
values (6921448, 184, 'Parachutes', 7822016);
```

1 KB	GD קובץ	22/05/2024 13:00	aircraft.gd 
1 KB	GD קובץ	21/05/2024 20:17	assigned_to.gd 
1 KB	GD קובץ	21/05/2024 19:24	base.gd 
2 KB	GD קובץ	21/05/2024 20:09	flight.gd 
1 KB	GD קובץ	21/05/2024 20:15	flown_by.gd 
1 KB	GD קובץ	21/05/2024 19:59	mission.gd 
1 KB	GD קובץ	21/05/2024 19:30	soldier.gd 
1 KB	GD קובץ	21/05/2024 20:12	supply.gd 

Aircraft.csv

Assigned_To.csv

Base.csv

Flight.csv

Flown_By.csv

Mission.csv

Soldier.csv

Supply.csv

Data from Textfile | Data to Oracle

File Data

307752,2626,Sde Dov Airbase
359439,5854,Sde Dov Airbase
746631,8085,Ramat David Airbase
958503,5938,Ovda Airbase

Configuration

General

Fieldcount
3

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

Quote character

"

Comment line

Import lines

1 ..

Field1 (+0 - ",")
Field2 (+0 - ",")
Field3 (+0 - ",")

Field Start

☐ Relative position
☐ Absolute position
☐ Character

Field End

☐ Length
☐ Character

Filter

Result Preview

1	2	3
307752	2626	Sde Dov Airbase
359439	5854	Sde Dov Airbase
746631	8085	Ramat David Airbase
958503	5938	Ovda Airbase

Data from Textfile | Data to Oracle

General

Owner

Table

SOLDIER

Initiali

Commit every...

100

☒ Overwrite duplicates
☐ Ignore duplicates

☐ Delete records
☐ Truncate table

Finaliz

Fields

Field1 -> PERSONNEL_ID (NUMBER)
Field2 -> RANK (VARCHAR2)
Field3 -> NAME_OF_SOLDIER (CHAR)

גיבוי

Export tables בעזרת sql/sql*

Name	Type	Compiled
AIRCRAFT	TABLE	5/27/2024 3:29:28 PM
AQ\$_INTERNET_AGENTS	TABLE	9/28/2021 4:39:51 AM
AQ\$_INTERNET_AGENT_PRIVS	TABLE	9/28/2021 4:39:51 AM
AQ\$_KEY_SHARD_MAP	TABLE	9/28/2021 4:39:47 AM
AQ\$_QUEUES	TABLE	9/28/2021 4:42:54 AM
AQ\$_QUEUE_TABLES	TABLE	9/28/2021 4:39:47 AM
AQ\$_SCHEDULES	TABLE	9/28/2021 4:39:48 AM
ASSIGNED_TO	TABLE	5/27/2024 3:29:28 PM
BASE	TABLE	5/27/2024 3:29:28 PM
FLIGHT	TABLE	5/27/2024 3:29:28 PM
FLOWN_BY	TABLE	5/27/2024 3:29:28 PM
HELP	TABLE	9/28/2021 6:41:11 AM
LOGMNR_CONSOL_CS	TABLE	9/28/2021 4:39:53 AM

User: <CURRENT USER>

Oracle Export | SQL Inserts | PL/SQL Developer

☐ Drop tables
☒ Create tables
☐ Truncate tables
☐ Delete records
☐ Disable triggers

☐ Disable foreign key constraints
☐ Include storage
☐ Include privileges

Commit every 400 records (0 = never)

☐ Zip Where clause

Output file: C:\Users\orinesher\Documents\db git repo\level 1\backup_27_5_2024.sql

Export

```
backup_27_5_2024.sql x
C:\Users\orinesher\Documents> db git repo > level 1 > backup_27_5_2024.sql

19  {
20  }
21  ;
22  alter table AIRCRAFT
23  add primary key (AIRCRAFT_ID);
24  alter table AIRCRAFT
25  add foreign key (BASE_ID)
26  references BASE (BASE_ID);
27
28  prompt Creating MISSION...
29  create table MISSION
30  (
31  mission_id    INTEGER not null,
32  date_of_mission DATE not null,
33  type         VARCHAR2(25) not null
34  )
35  ;
36  alter table MISSION
37  add primary key (MISSION_ID);
38
39  prompt Creating ASSIGNED_TO...
40  create table ASSIGNED_TO
41  (
42  aircraft_id  INTEGER not null,
43  mission_id   INTEGER not null
44  )
45  ;
46  alter table ASSIGNED_TO
47  add primary key (AIRCRAFT_ID, MISSION_ID);
48  alter table ASSIGNED_TO
49  add foreign key (AIRCRAFT_ID)
50  references AIRCRAFT (AIRCRAFT_ID);
51  alter table ASSIGNED_TO
52  add foreign key (MISSION_ID)
53  references MISSION (MISSION_ID);
54
55  
```

גיבוי לגיט

github.com/Ori1Nesher/DBProject_5506_4399/tree/master/level%201

ליק קודש יצירה תחביבים Dreemz

DBProject_5506_4399 / level 1 /

oriNesher final version all files

Name	Last commit message
..	
DGFiles	final version all files
DataImporterFiles	final version all files
insert	final version all files
mockarooFiles	final version all files
DSD.pdd	final version all files
DSD.~pdd	final version all files
ERD.png	final version all files
backup22.5.2024.sql	final version all files
createTable.sql	final version all files
dropTables.sql	final version all files
insertTables.sql	final version all files
selectAll.sql	final version all files

```
C:\Users\orinesher\Documents\db git repo>git add
warning: in the working copy of 'level 1/mockarooFiles/Solider.sql', LF will be replaced by CR
C:\Users\orinesher\Documents\db git repo>git commit -m "final version all files"
[master c0a4969] final version all files
33 files changed, 21132 insertions(+), 6805 deletions(-)
create mode 100644 level 1/DGFiles/aircraft.gd
create mode 100644 level 1/DGFiles/assigned_to.gd
create mode 100644 level 1/DGFiles/base.gd
create mode 100644 level 1/DGFiles/flight.gd
create mode 100644 level 1/DGFiles/flown_by.gd
create mode 100644 level 1/DGFiles/mission.gd
create mode 100644 level 1/DGFiles/soldier.gd
create mode 100644 level 1/DGFiles/supply.gd
create mode 100644 level 1/DSD.pdd
create mode 100644 level 1/DSD.~pdd
create mode 100644 level 1/DataImporterFiles/Soldier.csv
create mode 100644 level 1/ERD.png
create mode 100644 level 1/backup22.5.2024.sql
delete mode 100644 level 1/backup_20_5_2024.sql/aircraft_backup.sql
delete mode 100644 level 1/backup_20_5_2024.sql/base_backup.sql
delete mode 100644 level 1/backup_20_5_2024.sql/flight_backup.sql
delete mode 100644 level 1/backup_20_5_2024.sql/mission_backup.sql
delete mode 100644 level 1/backup_20_5_2024.sql/soldier_backup.sql
delete mode 100644 level 1/backup_20_5_2024.sql/supply_backup.sql
create mode 100644 level 1/insert/aircraft.sql
rename level 1/backup_20_5_2024.sql/assigned_to_backup.sql => insert/assigned_to.sql} (53%)
create mode 100644 level 1/insert/base.sql
create mode 100644 level 1/insert/flight.sql
rename level 1/backup_20_5_2024.sql/flown_by_backup.sql => insert/flown_by.sql} (52%)
create mode 100644 level 1/insert/mission.sql
create mode 100644 level 1/insert/soldier.sql
create mode 100644 level 1/insert/supply.sql
create mode 100644 level 1/insertTables.sql
create mode 100644 level 1/mockarooFiles/Solider.sql
create mode 100644 level 1/selectAll.sql
delete mode 100644 "level 1/\327\223\327\225\327\227 \327\224\327\244\327\250\327\225\327\231"
```

```
C:\Users\orinesher\Documents\db git repo>git push --force origin master
Enumerating objects: 55, done.
Counting objects: 100% (55/55), done.
Delta compression using up to 12 threads
Compressing objects: 100% (50/50), done.
Writing objects: 100% (55/55), 1.91 MiB | 167.00 KiB/s, done.
Total 55 (delta 16), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (16/16), done.
To https://github.com/Ori1Nesher/DBProject_5506_4399.git
+ e719379...c0a4969 master -> master (forced update)
```

שחזור בעזרת import tables

```
select * from FLIGHT t
dro

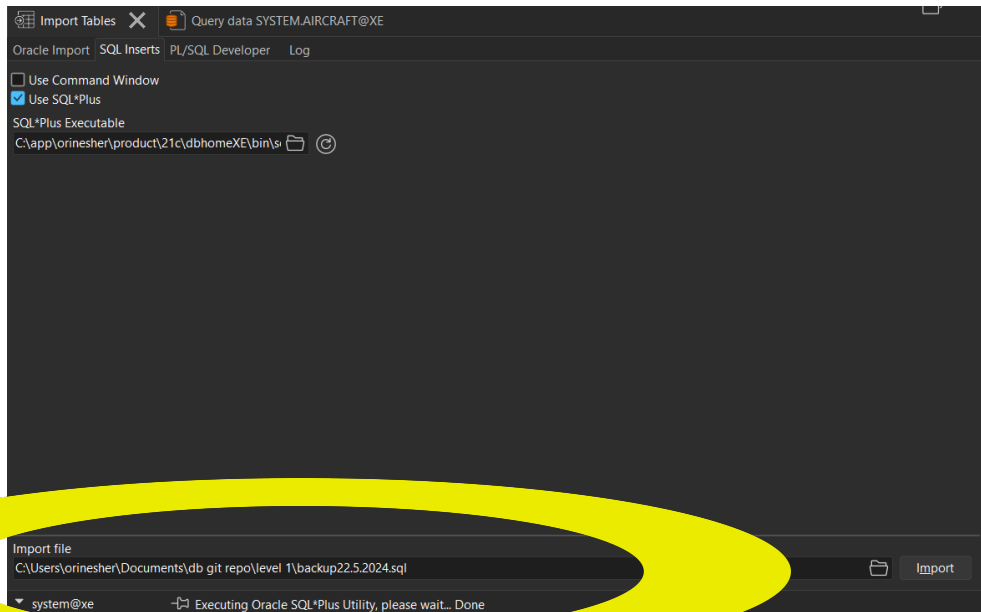
Oracle Import SQL Inserts PL/SQL D

Table created.

Table created.

Loading BASE...
400 records loaded
BASE DONE
400 records loaded
SOLDIER DONE
400 records loaded
MISSION DONE
Loading SUPPLY...
400 records loaded
SUPPLY DONE
Loading AIRCRAFT...
400 records loaded
AIRCRAFT DONE
Loading FLIGHT...
400 records loaded
FLIGHT DONE
Loading ASSIGNED TO...
400 records loaded
ASSIGNED TO DONE
Loading FLOWN_BY...
all tables are DONE

Import finished on 5/26/2
```



SQL Output Statistics

```
select * from FLIGHT t
```

	FLIGHT_ID	MISSION_ID	DATE_OF_FLIGHT	TYPE_OF_FLIGHT
1	3092399	5258982	1/28/2025	Cargo flights
2	1593812	3746189	9/23/2025	Cargo flights
3	6784749	3336293	12/14/2026	Aircraft ferrying flights
4	3175838	9092219	1/3/2001	Cargo flights
5	8881247	9919811	9/1/2005	Tactical transport flight
6	3806929	4669174	5/10/2014	Logistics support flights
7	4800948	8543360	8/30/2030	Weather reconnaissance
8	7359474	2234384	5/15/2018	Test flights
9	8622816	2702735	11/15/2016	Pilot proficiency flights
10	4645865	9991224	7/28/2030	Pilot proficiency flights
11	1177337	3094854	3/31/2008	Aircraft ferrying flights
12	7432167	5031926	11/23/2021	Calibration flights
13	2591449	7009557	12/30/2008	Surveillance flights
14	2014291	8565449	12/25/2031	Weather reconnaissance
15	1395956	1597631	9/10/2005	Aerial photography
16	2356668	7290296	4/13/2008	Aero-medical transport

הדרן עלך שלב א 😊

שלב ב

שאלות בחירה ללא פרמטרים

שליפת בסיסים צבאיים עם קיבולת מעל הממוצע

תיאור

השאלתה מיועדת לשלוף את כל הבסיסים צבאיים שהקיבולת שלהם גבוהה מהממוצע. בנוסף, היא מציגה את מספר סוגי האספקה השונים בכל בסיס. התוצאות מסודרות לפי מיקום הבסיסים.

הקוד

```
-- Retrieve Military Bases with Above-Average Capacities:
SELECT
    b.Base_ID,
    b.Location,
    b.Capacity,
    (SELECT AVG(b2.Capacity) FROM Base b2) AS Average_Capacity,
    (SELECT COUNT(DISTINCT s.Supply_Name)
     FROM Supply s
     WHERE s.Base_ID = b.Base_ID) AS Number_Of_Supply_Types
FROM
    Base b
WHERE
    b.Capacity > (SELECT AVG(b2.Capacity) FROM Base b2)
ORDER BY
    b.Location;
```

תוצאות

	BASE_ID	LOCATION	CAPACITY	AVERAGE_CAPACITY	NUMBER_OF_SUPPLY_TYPES
1	5105795	North	8982	5984.9275	0
2	1118980	North	6278	5984.9275	1
3	8680745	North	8032	5984.9275	0
4	6343849	North	9984	5984.9275	0
5	3879408	North	7356	5984.9275	2
6	4449294	North	7245	5984.9275	0
7	8269150	North	7501	5984.9275	1
8	1783223	North	9394	5984.9275	1

ספירת טיסות לפי סוג משימה

תיאור

השאלתה סופרת את מספר הטיסות עבור כל סוג משימה ומציגה את התוצאות בסדר יורד לפי מספר הטיסות.

הקוד

```
-- Count Flights per Mission Type:
SELECT
    m.Type AS Mission_Type,
    COUNT(f.Flight_ID) AS Number_Of_Flights
FROM
    Mission m
LEFT JOIN
    Flight f ON m.Mission_ID = f.Mission_ID
GROUP BY
    m.Type
ORDER BY
    Number_Of_Flights DESC;
```

תוצאות

		MISSION_TYPE		NUMBER_OF_FLIGHTS	
▶	1	Air interdiction	***	98	
	2	Close air support	***	90	
	3	Tactical bombing	***	82	
	4	Air superiority	***	71	
	5	Strategic bombing	***	59	

רשימת משימות מחמש השנים האחרונות לפי סוג מטוס

תיאור

השאלתה שולפת משימות מהחמש השנים האחרונות ומציגה אותן מקובצות לפי סוג המטוס וסוג המשימה. התוצאות מסודרות לפי תאריך המשימה, סוג המשימה וסוג המטוס.

הקוד

```
-- List Missions from the Last Five Years Grouped by Aircraft Type:
SELECT
    m.Mission_ID,
    m.Type AS Mission_Type,
    m.Date_Of_Mission,
    a.Type AS Aircraft_Type,
    COUNT(*) OVER (PARTITION BY m.Type, a.Type) AS Number_Of_Missions
FROM
    Mission m
JOIN
    Assigned_To at ON m.Mission_ID = at.Mission_ID
JOIN
    Aircraft a ON at.Aircraft_ID = a.Aircraft_ID
WHERE
    m.Date_Of_Mission >= ADD_MONTHS(SYSDATE, -60)
    AND m.Date_Of_Mission <= SYSDATE
ORDER BY
    m.Date_Of_Mission, m.Type, a.Type;
```

תוצאות

	MISSION_ID	MISSION_TYPE	DATE_OF_MISSION	AIRCRAFT_TYPE	NUMBER_OF_MISSIONS
1	2433761	Strategic bombing	14/06/2019	F-15 Baz	7
2	2433761	Strategic bombing	14/06/2019	F-15 Baz	7
3	2433761	Strategic bombing	14/06/2019	F-15I Raam	7
4	6597242	Tactical bombing	31/08/2019	F-15 Baz	4
5	4084827	Close air support	05/09/2019	F-35I Adir	5
6	5354867	Air interdiction	01/01/2020	F-15 Baz	2

כל הטיסות מאותו סוג של הטיסה הבאה שתתרחש

תיאור

השאלתה מציגה את הטיסה הקרובה ביותר בתאריך לתאריך הנוכחי, ואת כל הטיסות מאותו סוג. התוצאות ממיינות לפי תאריך הטיסה בסדר עולה.

הקוד

```
--The next flight and all flights of the same type that did not occur in even-numbered months.
WITH Next_Flight AS (
    SELECT
        f.Flight_ID,
        f.Mission_ID,
        f.Date_Of_Flight,
        f.Type_Of_Flight
    FROM
        Flight f
    WHERE
        f.Date_Of_Flight = (SELECT MIN(f1.Date_Of_Flight)
                           FROM Flight f1
                           WHERE f1.Date_Of_Flight > SYSDATE)
)
SELECT
    nf.Flight_ID,
    nf.Mission_ID,
    nf.Date_Of_Flight,
    nf.Type_Of_Flight
FROM
    Next_Flight nf
UNION ALL
SELECT
    f.Flight_ID,
    f.Mission_ID,
    f.Date_Of_Flight,
    f.Type_Of_Flight
FROM
    Flight f
WHERE
    f.Type_Of_Flight = (SELECT nf.Type_Of_Flight FROM Next_Flight nf)
ORDER BY
    Date_Of_Flight;
```

תוצאות

		FLIGHT_ID	MISSION_ID	DATE_OF_FLIGHT	TYPE_OF_FLIGHT	
▶	1	9913430	1071270	07/03/2000	Non-combat reconnaissance	***
	2	2475897	1348858	01/05/2000	Non-combat reconnaissance	***
	3	4506892	7160188	26/07/2000	Non-combat reconnaissance	***
	4	9816358	7118731	18/04/2001	Non-combat reconnaissance	***
	5	4000533	2488640	25/06/2001	Non-combat reconnaissance	***
	6	4143547	2897777	11/08/2001	Non-combat reconnaissance	***
	7	9668411	5395552	19/11/2001	Non-combat reconnaissance	***

שאלות עם פרמטרים

כל המטוסים עם סטטוס מסוים ומיקום מסוים

תיאור

השאלתה מחזירה את מזהה המטוס והסוג שלו בהתאם לסטטוס מסוים בבסיס צבאי מסוים, הממוקם באזור הדרום או הצפון. המשתמש יזין את הסטטוס הרצוי למטוס (פעיל, לא פעיל, תחזוקה) ואת המיקום הרצוי של הבסיס (דרום או צפון).

הקוד

```
--Retrieve the Aircraft ID and Type of aircraft with a specified
--status at a particular military base location, either in the South or North

SELECT a.Aircraft_ID, a.Type
FROM Aircraft a
JOIN Base b ON a.Base_ID = b.Base_ID
WHERE a.Status = &<name = "status" type = "string" list = "Active,Inactive,Under Maintenance"
| hint = "Enter the status of the aircraft">
AND b.Location = &<name = "base_location" type = "string" list = "South,North" hint = "Enter the base location">;
```

תוצאות

	AIRCRAFT_ID	TYPE
1	8582823	F-35I Adir
2	4064250	F-15I Raam
3	2346062	F-35I Adir
4	6605609	F-15I Raam
5	2865561	F-15 Baz
6	3910321	F-15 Baz
7	1629105	F-35I Adir
8	5062504	F-15 Baz

Variables

Name	Value
status	Inactive
base_location	North
	South
	North

Enter the base location

OKCancelClear

כל החיילים שטסו בשנה מסוימת במשימה מסוג מסוים

תיאור

השאלתה הזו בוחרת את כל החיילים שטסו בשנה מסוימת והשתתפו במשימה מסוג מסוים. המשתמש מזין את השנה של הטיסה ואת סוג המשימה הנדרשת מתוך רשימת סוגי המשימות האפשריות.

הקוד

```
-- Selects soldiers who flew in a specific year and participated in missions of a specific type.
SELECT s.Personnel_ID, s.Name_Of_Soldier, s.Rank
FROM Soldier s
JOIN Flown_By fb ON s.Personnel_ID = fb.Personnel_ID
JOIN Aircraft a ON fb.Aircraft_ID = a.Aircraft_ID
JOIN Assigned_To at ON a.Aircraft_ID = at.Aircraft_ID
JOIN Mission m ON at.Mission_ID = m.Mission_ID
JOIN Flight f ON m.Mission_ID = f.Mission_ID
WHERE EXTRACT(YEAR FROM f.Date_Of_Flight) = &<name="Flight Year" hint="Enter flight year as YYYY" type="integer">
AND m.Type = &<name="Mission Type" hint="Enter the type of mission" type="string"
list = "Air superiority, Air interdiction, Close air support, Strategic bombing, Tactical bombing";
```

תוצאות

Variables

Name	Value
Flight Year	2020
Mission Type	Strategic bombing

Enter flight year as YYYY

OK

Cancel

Clear

	PERSONNEL_ID	NAME_OF_SOLDIER	RANK
1	5300630	Greg	Aluf Mishne
2	5542663	Mykelti	Rav Seren
3	1414877	Cloris	Aluf Mishne
4	2507386	Merrilee	Sgan Aluf

משימות שבוצעו על ידי חיילים עם דרגה מסוימת בשנתיים האחרונות

תיאור

השאלתה מחזירה את מזהה המשימה, תאריך המשימה, סוג המשימה ושם החיילים שביצעו אותה, עבור משימות שבוצעו על ידי חיילים של דרגה מסוימת בשנתיים האחרונות. המשתמש יזין את הדרגה הרצויה של החיילים.

הקוד

```
--Missions conducted by soldiers of a specific rank within the past two years.

SELECT m.Mission_ID, m.Date_Of_Mission, m.Type AS Mission_Type, s.Name_Of_Soldier
FROM Mission m
JOIN Assigned_To at ON m.Mission_ID = at.Mission_ID
JOIN Aircraft a ON at.Aircraft_ID = a.Aircraft_ID
JOIN Flown_By fb ON a.Aircraft_ID = fb.Aircraft_ID
JOIN Soldier s ON fb.Personnel_ID = s.Personnel_ID
WHERE s.Rank = &<name = "rank" type = "string" list = "Seren, Rav Seren, Sgan Aluf, Aluf Mishne"
hint = "Enter the rank of the soldier">
AND m.Date_Of_Mission BETWEEN ADD_MONTHS(TRUNC(SYSDATE), -24) AND TRUNC(SYSDATE)
ORDER BY m.Date_Of_Mission DESC;
```

תוצאות

Variables

Name	Value
rank	Aluf Mishne

Seren
Rav Seren
Sgan Aluf
Aluf Mishne

Enter the rank of the soldier

OK Cancel Clear

	MISSION_ID	DATE_OF_MISSION	MISSION_TYPE	NAME_OF_SOLDIER
1	8600140	10/04/2024	Close air support	Miko
2	1062224	13/03/2024	Air superiority	Fred
3	1062224	13/03/2024	Air superiority	Rueben
4	5612097	21/07/2023	Strategic bombing	Debra
5	9748374	05/06/2023	Strategic bombing	Owen
6	8010012	04/12/2022	Air superiority	Mvkelti

פרטי חיילים, משימות, מטוסים ובסיסים על פי דרגת החייל, מיקום הבסיס וקיבולת מינימלית של הבסיס

תיאור

השאלתה מחזירה את פרטי החייל, המשימה, המטוס והבסיס בהתאם לדרגת החייל, מיקום הבסיס (צפון או דרום) ולקיבולת המינימלית של הבסיס. המשתמש יזין את הדרגה הרצויה של החייל, את מיקום הבסיס (צפון או דרום) ואת הקיבולת המינימלית של הבסיס.

הקוד

```
--retrieves soldier, mission, aircraft, and base information based on rank, location, and capacity filters.

SELECT S.Personnel_ID, S.Name_Of_Soldier, S.Rank, M.Mission_ID,
M.Date_Of_Mission, M.Type, A.Type AS Aircraft_Type, A.Status, B.Location, B.Capacity
FROM Soldier S
JOIN Flown_By FB ON S.Personnel_ID = FB.Personnel_ID
JOIN Aircraft A ON FB.Aircraft_ID = A.Aircraft_ID
JOIN Assigned_To AT ON A.Aircraft_ID = AT.Aircraft_ID
JOIN Mission M ON AT.Mission_ID = M.Mission_ID
JOIN Base B ON A.Base_ID = B.Base_ID
WHERE S.Rank = &<name="Rank" hint="Rank of the soldier" List = "Seren, Rav Seren, Sgan Aluf, Aluf Mishne" type="string">
AND B.Location = &<name="Location" hint="Location of the base" List = "North, South" type="string">
AND B.Capacity > &<name="Min_Capacity" hint="Minimum capacity of the base" type="integer">;
```

תוצאות

Variables

	Name	Value
✓	Rank	Sgan Aluf
	Location	South
	Min_Capacity	700

Rank of the soldier

OKCancelClear

	PERSONNEL_ID	NAME_OF_SOLDIER	RANK	MISSION_ID	DATE_OF_MISSION	TYPE	AIRCRAFT_TYPE	STATUS	LOCATION	CAPACITY
▶ 1	9617099	Ty	Sgan Aluf	1623369	14/05/2008	Strategic bombing	F-35I Adir	Inactive	South	9292
2	9617099	Ty	Sgan Aluf	1138784	09/10/2030	Air interdiction	F-35I Adir	Inactive	South	9292
3	8075961	Gladys	Sgan Aluf	7814021	06/05/2020	Air interdiction	F-35I Adir	Under Maintenance	South	2565
4	8075961	Gladys	Sgan Aluf	5204185	27/03/2002	Air superiority	F-35I Adir	Under Maintenance	South	2565
5	7484796	Uma	Sgan Aluf	1216187	23/11/2014	Strategic bombing	F-15I Raam	Inactive	South	9551

שאלות מחיקה

הוצאת סוג כלי טיס משימוש

תיאור

השאלתה היא משפט DELETE שנועד להסיר רשומות ספציפיות מהטבלה Assigned_To. הרשומות המיועדות למחיקה נקבעות על סמך קריטריונים הכוללים שתי טבלאות אחרות: מטוס ומשימה. השאלתה תמחק את כל הרשומות של משימות עתידיות, שרשום להן כלי טיס מסוג "C-130 Hercules Karnaf". בעצם מדמה סיטואציה בה סוג מסוים של כלי טיס הוצא מכלל שימוש.

קוד

```
DELETE FROM Assigned_To
WHERE Aircraft_ID IN (
    SELECT Aircraft_ID
    FROM Aircraft
    WHERE Type = 'C-130 Hercules Karnaf'
)
AND Mission_ID IN (
    SELECT Mission_ID
    FROM Mission
    WHERE Date_Of_Mission > SYSDATE
);
commit;
```

תוצאות

התמונה כאן מראה את כל הרשומות שהשאלתה אמורה למחוק. (6 רשומות)

```
SELECT *
FROM Assigned_To
WHERE Aircraft_ID IN (
    SELECT Aircraft_ID
    FROM Aircraft
    WHERE Type = 'C-130 Hercules Karnaf'
)
AND Mission_ID IN (
    SELECT Mission_ID
    FROM Mission
    WHERE Date_Of_Mission > SYSDATE
);
```

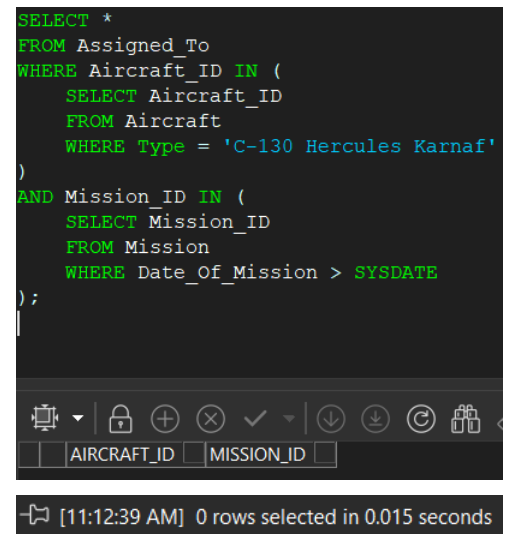
	AIRCRAFT_ID	MISSION_ID
1	1480278	2709369
2	1481278	2709369
3	1482278	2709369
4	1483278	2709369
5	1484278	2709369
6	1485278	2709369

נריץ את שאלת המחיקה שלנו:

[11:11:40 AM] 6 rows deleted in 0.020 seconds

כאן אפשר לראות שלאחר המחיקה השאילתה שבוחרת את כל הרשומות ששאילתת המחיקה אמורה למחוק לא מוצאת כלום.

```
SELECT *
FROM Assigned_To
WHERE Aircraft_ID IN (
  SELECT Aircraft_ID
  FROM Aircraft
  WHERE Type = 'C-130 Hercules Karnaf'
)
AND Mission_ID IN (
  SELECT Mission_ID
  FROM Mission
  WHERE Date_Of_Mission > SYSDATE
);
```



[11:12:39 AM] 0 rows selected in 0.015 seconds

מחיקת משימה

תיאור

המטרה היא להסיר לחלוטין את כל העקבות של משימה ספציפית ממסד הנתונים. הדבר כולל מחיקת ערכים ממספר טבלאות המתייחסות למזהה המשימה, תוך הבטחה שלא יישארו מאחור רשומות מיותרות. הטבלאות המעורבות הן Assigned_To, Flight ו-Mission.

קוד

```
DELETE FROM Assigned_To
WHERE Mission_ID = 3218552;

DELETE FROM Flight
WHERE Mission_ID = 3218552;

DELETE FROM Mission
WHERE Mission_ID = 3218552;

COMMIT;
```

תוצאות

זאת הרשומה שנרצה למחוק:

MISSION_ID	DATE_OF_MISSION	TYPE
2709369	7/13/2027	Tactical bombing

נריץ את השאילתה שלנו:

```
DELETE FROM Assigned_To
WHERE Mission_ID = 2709369;

DELETE FROM Flight
WHERE Mission_ID = 2709369;

DELETE FROM Mission
WHERE Mission_ID = 2709369;

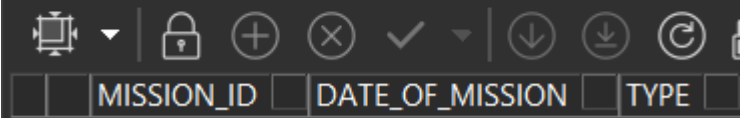
COMMIT;
```

1 row deleted in 0.002 seconds Delete assigned_to

2 rows deleted in 0.003 seconds Delete flight

1 row deleted in 0.004 seconds Delete mission

```
select *  
from Mission  
where mission_id = 2709369
```



ניתן לראות בהרצת השאילתה הזאת שאין יותר משימה עם מפתח כמו זה שמחקנו בטבלה של mission.

שאלות עדכון

תיאור

מטרת שאלתה זו היא לעדכן את מצב המטוסים שהוקצו למשימות המתוכננות להתרחש בשנה הקרובה (מהתאריך הנוכחי) בטבלת המטוסים ל- 'preparation'.

קוד

```
UPDATE Aircraft
SET Status = 'preparation'
WHERE Aircraft_ID IN (
    SELECT Aircraft.Aircraft_ID
    FROM Aircraft
    JOIN Assigned_To ON Aircraft.Aircraft_ID = Assigned_To.Aircraft_ID
    JOIN Mission ON Assigned_To.Mission_ID = Mission.Mission_ID
    WHERE Mission.Date_Of_Mission BETWEEN SYSDATE AND ADD_MONTHS(SYSDATE, 12)
);|
```

תוצאות

10 rows updated in 0.030 seconds

```
select *
from aircraft
where status = 'preparation'
```

		AIRCRAFT_ID	TYPE		STATUS		BASE_ID
▶	1	9866928	F-35I Adir	---	preparation	---	5648560
	2	9860541	F-15 Baz	---	preparation	---	9194052
	3	7725207	F-15 Baz	---	preparation	---	4253827
	4	4121496	F-16I Sufa	---	preparation	---	8802505
	5	5475461	F-15I Raam	---	preparation	---	9019860
	6	9826361	F-35I Adir	---	preparation	---	5858610
	7	2880930	F-15I Raam	---	preparation	---	3329277
	8	2419883	F-15 Baz	---	preparation	---	5159719
	9	1776720	F-15 Baz	---	preparation	---	8242911
	10	7725614	F-15I Raam	---	preparation	---	5278329

תיקון מטוס

תיאור

המטרה של השאילתה היא לעדכן את רישומי המלאי של מנועים בטבלת Supply, באופן ספציפי עבור בסיסים שבהם מטוסים נמצאים בתהליך תחזוקה. השאילתה מפחיתה את כמות המנועים ב-10 עבור כל רישום רלוונטי. המטרה הייתה ליצור איזשהו אב טיפוס לשאילתה שתעדכן את נתוני האספקה של הבסיס לאחר תיקון של מטוס מסוים.

קוד

```
UPDATE Supply
SET Quantity = Quantity - 10
WHERE Supply_Name = 'Engines'
AND Base_ID IN (
    SELECT Base_ID
    FROM Aircraft
    WHERE Status = 'Under Maintenance'
);
commit
```

תוצאות

הנה שאילתת שבועות את כל הרשומות בשאילתת העדכון נוגעת בהן לפני ואחרי הרצת שאילתת העדכון

	SUPPLY_ID	QUANTITY	SUPPLY_NAME	BASE_ID
1	9282469	404	Engines	1719804
2	9592256	598	Engines	5663092
3	1760055	894	Engines	7588771
4	8639709	661	Engines	9675914
5	2550032	670	Engines	5079970
6	8688982	201	Engines	7530416
7	5206275	873	Engines	7783391
8	2390860	761	Engines	5303110
9	8530787	470	Engines	8005820
10	8642879	136	Engines	6938609
11	2588495	664	Engines	3292604
12	8500002	1000	Engines	5347409
13	4927951	125	Engines	4299184
14	8043871	727	Engines	9859836
15	5728111	599	Engines	9859836
16	6685478	530	Engines	8934054
17	8538738	47	Engines	8934054
18	4496485	184	Engines	4491012

	SUPPLY_ID	QUANTITY	SUPPLY_NAME	BASE_ID
1	9282469	414	Engines	1719804
2	9592256	608	Engines	5663092
3	1760055	904	Engines	7588771
4	8639709	671	Engines	9675914
5	2550032	680	Engines	5079970
6	8688982	211	Engines	7530416
7	5206275	883	Engines	7783391
8	2390860	771	Engines	5303110
9	8530787	480	Engines	8005820
10	8642879	146	Engines	6938609
11	2588495	674	Engines	3292604
12	8500002	1010	Engines	5347409
13	4927951	135	Engines	4299184
14	8043871	737	Engines	9859836
15	5728111	609	Engines	9859836
16	6685478	540	Engines	8934054
17	8538738	57	Engines	8934054
18	4496485	194	Engines	4491012

ניתן לראות שירד עשר מ-Quantity

הוספת אילוצים

תיאור

- הוספת הגבלת בדיקה לטבלת 'Mission' שמוודאת כי סוג המשימה הינו אחד מהערכים שהגדרנו מראש.
 - הוספת ערך ברירת מחדל (הערך הינו 'Active') לעמודה 'Status' בטבלת 'Aircraft'.
 - הוספת מגבלת ייחודיות לעמודה 'Date_Of_Flight' בטבלת 'Flight', כך שתאריך הטיסה יהיה ייחודי.
- (לאחר מכן, הקוד מאשר את השינויים עם הפקודה 'COMMIT').

קוד

```
ALTER TABLE Mission
ADD CONSTRAINT chk_type CHECK (
    Type IN (
        'Air superiority', 'Air interdiction', 'Close air support', 'Strategic bombing', 'Tactical bombing'
    )
);

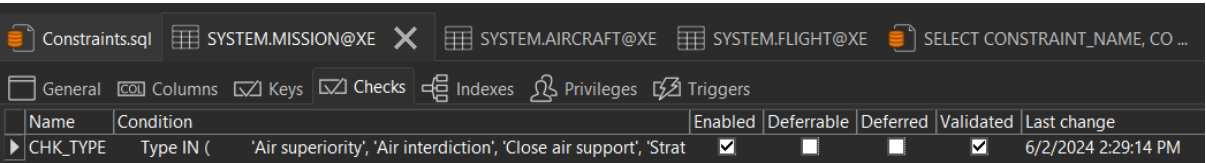
ALTER TABLE AIRCRAFT
MODIFY Status VARCHAR2(25) DEFAULT 'Active';

ALTER TABLE Flight
ADD CONSTRAINT unique_date_of_flight UNIQUE (Date_Of_Flight);

commit;
```

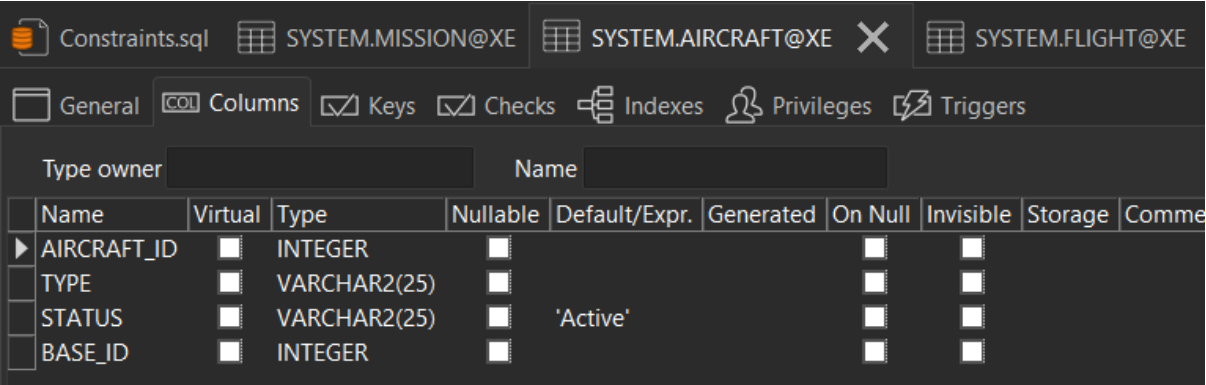
תוצאות

לאחר שהרצנו את התוכנית:
ניתן לראות בתמונה הבאה את הבדיקה ב-mission:



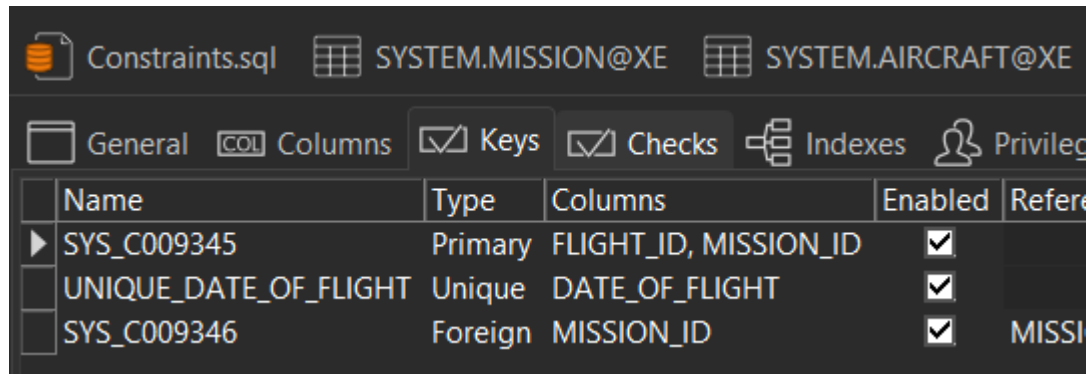
Name	Condition	Enabled	Deferrable	Deferred	Validated	Last change
CHK_TYPE	Type IN ('Air superiority', 'Air interdiction', 'Close air support', 'Strat	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	6/2/2024 2:29:14 PM

ניתן לראות בתמונה הבאה את הערך ברירת המחדל החדש בסטטוס של מטוס:



Name	Virtual	Type	Nullable	Default/Expr.	Generated	On Null	Invisible	Storage	Comme
AIRCRAFT_ID	<input type="checkbox"/>	INTEGER	<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>		
TYPE	<input type="checkbox"/>	VARCHAR2(25)	<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>		
STATUS	<input type="checkbox"/>	VARCHAR2(25)	<input type="checkbox"/>	'Active'		<input type="checkbox"/>	<input type="checkbox"/>		
BASE_ID	<input type="checkbox"/>	INTEGER	<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>		

ניתן לראות בתמונה הבאה את הבדיקה של התאריך של טיסה שיהיה "מיוחד"



Constraints.sql SYSTEM.MISSION@XE SYSTEM.AIRCRAFT@XE					
General Columns Keys Checks Indexes Privileges					
	Name	Type	Columns	Enabled	Refer
▶	SYS_C009345	Primary	FLIGHT_ID, MISSION_ID	<input checked="" type="checkbox"/>	
	UNIQUE_DATE_OF_FLIGHT	Unique	DATE_OF_FLIGHT	<input checked="" type="checkbox"/>	
	SYS_C009346	Foreign	MISSION_ID	<input checked="" type="checkbox"/>	MISSION_ID

הדרן עלך שלב ב 😊

שלב ג

תוכנית א

הסבר

התוכנית שלנו מורכבת מפונקציה, פרוצדורה ותוכנית ראשית אשר יחד מעדכנים את דרגות החיילים ששמותיהם מתחילים באותיות מ- 'A' עד 'M' (החצי הראשון של ה-ABC).

הפונקציה **get_soldiers_first_half_alphabet** מחזירה cursor עם שמות החיילים הללו. כלומר: היא מחזירה את כל החיילים ששמותיהם הפרטיים נמצאים בחצי הראשון של ה-ABC - השמות שמתחילות באותיות בין A-M.

הפרוצדורה **update_soldiers_rank_from_cursor** מקבל את cursor של כל החיילים מהפונקציה ודרגה חדשה כקלט, ואז עוברת בלולאה על cursor כדי לעדכן את דרגתו של כל חייל ל- **Sergeant**.

התוכנית הראשית קוראת לפונקציה כדי לקבל את cursor ולאחר מכן קוראת לפרוצדורה בכדי לעדכן את הדרגות, ישנו טיפול במקרים חריגים כדי להבטיח שהcursor סגור והדפסת ההודעות הצליחה או נכשלה.

קוד

```
--FUNCTION
CREATE OR REPLACE FUNCTION get_soldiers_first_half_alphabet
RETURN SYS_REFCURSOR
IS
    rc SYS_REFCURSOR;
BEGIN
    OPEN rc FOR
        SELECT Name_Of_Soldier
        FROM Soldier
        WHERE UPPER(SUBSTR(Name_Of_Soldier, 1, 1)) BETWEEN 'A' AND 'M';

    RETURN rc;
EXCEPTION
    WHEN OTHERS THEN
        -- Handle the exception by closing the cursor and returning a message
        CLOSE rc;
        RAISE_APPLICATION_ERROR(-20001, 'Error occurred: ' || SQLERRM);
END;
/

--PROCEDURE
CREATE OR REPLACE PROCEDURE update_soldiers_rank_from_cursor(
    rc IN SYS_REFCURSOR,
    new_rank IN VARCHAR2
```

```

)
IS
    v_name Soldier.Name_Of_Soldier%TYPE;
BEGIN
    -- Loop through the ref cursor and update each soldier's rank
    LOOP
        FETCH rc INTO v_name;
        EXIT WHEN rc%NOTFOUND;

        -- Update the rank of the soldier
        UPDATE Soldier
        SET Rank = new_rank
        WHERE Name_Of_Soldier = v_name;
    END LOOP;

    -- Close the cursor
    CLOSE rc;

    -- Commit the transaction
    COMMIT;
END;
/

--MAIN PROGRAM
DECLARE
    rc SYS_REFCURSOR;
BEGIN
    -- Call the function to get the cursor
    rc := get_soldiers_first_half_alphabet;

    -- Call the procedure to update the soldiers' ranks
    update_soldiers_rank_from_cursor(rc, 'Sergeant');

    -- Print success message
    DBMS_OUTPUT.PUT_LINE('Successfully executed');
EXCEPTION
    WHEN OTHERS THEN
        -- Print error message
        DBMS_OUTPUT.PUT_LINE('Error occurred: ' || SQLERRM);
        -- Ensure the cursor is closed if an error occurs
        IF rc%ISOPEN THEN
            CLOSE rc;
        END IF;
        -- Re-raise the exception
        RAISE;
END;/

```

הרצה

לטובת ההרצה יצרנו תוכנית פשוטה שמדפיסה את כל החיילים אותם הפונקציה עתידה להכניס לסמן:
לפני ההרצה של התוכנית:

```
SELECT Name_Of_Soldier, rank
FROM Soldier
WHERE UPPER(SUBSTR(Name_Of_Soldier, 1, 1)) BETWEEN 'A' AND 'M'
ORDER BY Soldier.name_of_soldier;
```

	NAME_OF_SOLDIER	RANK
1	Adrien	Aluf Mishne
2	Alan	Sgan Aluf
3	Alannah	Sgan Aluf
4	Albertina	Rav Seren
5	Alfie	Rav Seren
6	Alfred	Rav Seren
7	Alice	Rav Seren
8	Alice	Aluf Mishne
9	Alicia	Seren
10	Andie	Sgan Aluf
11	Andre	Aluf Mishne
12	Andre	Aluf Mishne
13	Andrew	Rav Seren
14	Angela	Rav Seren
15	Ani	Aluf Mishne
16	Anita	Rav Seren
17	Anita	Sgan Aluf

אחרי ההרצה:

```
SELECT Name_Of_Soldier, rank
FROM Soldier
WHERE UPPER(SUBSTR(Name_Of_Soldier, 1, 1)) BETWEEN 'A' AND 'M'
ORDER BY Soldier.name_of_soldier;
```

	NAME_OF_SOLDIER	RANK
1	Adrien	Sergeant
2	Alan	Sergeant
3	Alannah	Sergeant
4	Albertina	Sergeant
5	Alfie	Sergeant
6	Alfred	Sergeant
7	Alice	Sergeant
8	Alice	Sergeant
9	Alicia	Sergeant
10	Andie	Sergeant
11	Andre	Sergeant
12	Andre	Sergeant
13	Andrew	Sergeant
14	Angela	Sergeant
15	Ani	Sergeant
16	Anita	Sergeant
17	Anita	Sergeant

תוכנית ב

הסבר:

התוכנית שלנו מורכבת מפונקציה, פרוצדורה ותכנית ראשית אשר ביחד מעדכנים את קיבולת הבסיסים בהתבסס על כמות האספקה שלהם.

הפונקציה **GetTopOrBottom100Bases** מחזירה cursor עם 100 הבסיסים בעלי כמות האספקה הגדולה ביותר או הקטנה ביותר, בהתאם לממוצע כמויות האספקה.

כלומר:

- אם ממוצע כמויות האספקה גדול ממספר 600 (בחרנו סתם מספר) אז היא תחזיר 100 הבסיסים בעלי כמות האספקה הגדולה ביותר (בסדר יורד).
- ואם ממוצע כמויות האספקה קטן ממספר 600 (בחרנו סתם מספר) אז היא תחזיר 100 הבסיסים בעלי כמות האספקה הקטנה ביותר (בסדר עולה).

הפרוצדורה **AdjustBaseCapacity** לוקחת את הסמן הזה כקלט, ואז מעדכנת את הקיבולת של כל בסיס לפי סדר הנתונים (עולה או יורד).

כלומר:

- אם הקלט בסדר יורד אז הממוצע כמויות האספקה גדול ולכן נעדכן את ה100 הבסיסים בתוספת של 50 לכמות האספקה הכוללת.
- אם הקלט בסדר עולה אז הממוצע כמויות האספקה קטן ולכן נעדכן את ה100 הבסיסים בהפחתה של 50 מכמות האספקה הכוללת.

המיין קורא לפונקציה כדי לקבל את הסמן ולאחר מכן קורא לעדכון הקיבולת, במצב של שגיאה אנחנו עושים ROLLBACK כלומר מבטלים את פעולת העדכון של הפרוצדורה. ואז מעלים את השגיאה מחדש כדי שהמערכת תקפיץ את השגיאה למשתמש.

קוד:

```
CREATE OR REPLACE FUNCTION GetTopOrBottom100Bases RETURN SYS_REFCURSOR
IS
    rc SYS_REFCURSOR;
    v_Avg_Supply NUMBER;
BEGIN
    -- Calculate the average supply quantity per base
    SELECT AVG(total_supply) INTO v_Avg_Supply
    FROM (
        SELECT SUM(Quantity) AS total_supply
        FROM Supply
        GROUP BY Base_ID
    );
    DBMS_OUTPUT.PUT_LINE('The Average is: ' || v_Avg_Supply);
    -- Determine if average is higher than 5
```

```

IF v_Avg_Supply > 600 THEN
    -- Return the 100 bases with the largest amount of supply
    OPEN rc FOR
    SELECT Base_ID, SUM(Quantity) AS total_supply
    FROM Supply
    GROUP BY Base_ID
    ORDER BY total_supply DESC
    FETCH FIRST 100 ROWS ONLY;
ELSE
    -- Return the 100 bases with the smallest amount of supply
    OPEN rc FOR
    SELECT Base_ID, SUM(Quantity) AS total_supply
    FROM Supply
    GROUP BY Base_ID
    ORDER BY total_supply ASC
    FETCH FIRST 100 ROWS ONLY;
END IF;

RETURN rc;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        OPEN rc FOR SELECT 'No Data Found' AS Message FROM DUAL;
        RETURN rc;
    WHEN OTHERS THEN
        OPEN rc FOR SELECT 'Error' AS Message FROM DUAL;
        RETURN rc;
END GetTopOrBottom100Bases;
/

CREATE OR REPLACE PROCEDURE AdjustBaseCapacity(p_rc IN SYS_REFCURSOR)
IS
    TYPE SupplyRec IS RECORD (
        Base_ID Base.Base_ID%TYPE,
        Total_Supply NUMBER
    );

    v_First_Supply SupplyRec;
    v_Last_Supply SupplyRec;
    v_Order VARCHAR2(10);

    -- Temporary table to store cursor data
    TYPE SupplyTable IS TABLE OF SupplyRec;
    v_SupplyTable SupplyTable := SupplyTable();

```

```

BEGIN
    -- Fetch all rows from the cursor into the collection
    LOOP
        FETCH p_rc INTO v_First_Supply.Base_ID,
v_First_Supply.Total_Supply;
        EXIT WHEN p_rc%NOTFOUND;
        v_SupplyTable.EXTEND;
        v_SupplyTable(v_SupplyTable.COUNT) := v_First_Supply;
    END LOOP;
    CLOSE p_rc;

    -- Determine the order based on the first and last rows
    v_First_Supply := v_SupplyTable(1);
    v_Last_Supply := v_SupplyTable(v_SupplyTable.COUNT);

    IF v_First_Supply.Total_Supply > v_Last_Supply.Total_Supply THEN
        v_Order := 'DESC';
    ELSE
        v_Order := 'ASC';
    END IF;

    -- Adjust the capacity based on the determined order
    IF v_Order = 'DESC' THEN
        FOR i IN v_SupplyTable.FIRST..v_SupplyTable.LAST LOOP
            UPDATE Base
            SET Capacity = Capacity + 50
            WHERE Base_ID = v_SupplyTable(i).Base_ID;
        END LOOP;
    ELSE
        FOR i IN v_SupplyTable.FIRST..v_SupplyTable.LAST LOOP
            UPDATE Base
            SET Capacity = Capacity - 50
            WHERE Base_ID = v_SupplyTable(i).Base_ID;
        END LOOP;
    END IF;

    -- Commit the changes
    COMMIT;
EXCEPTION
    WHEN OTHERS THEN
        ROLLBACK;
        RAISE;
END AdjustBaseCapacity;

```

```

/

DECLARE
    rc SYS_REFCURSOR;
BEGIN
    -- Call the GetTopOrBottom100Bases function
    rc := GetTopOrBottom100Bases;

    -- Call the AdjustBaseCapacity procedure
    AdjustBaseCapacity(rc);

    DBMS_OUTPUT.PUT_LINE('Base capacities adjusted successfully.');
```

END;

/

הרצה

יצרנו select פשוט שיעזור להראות את ההבדלים לפני ואחרי הרצת התוכנית:

```

SELECT s.Base_ID, s.total_supply, b.Capacity
FROM (
    SELECT Base_ID, SUM(Quantity) AS total_supply
    FROM Supply
    GROUP BY Base_ID
) s
JOIN Base b ON s.Base_ID = b.Base_ID
ORDER BY s.total_supply DESC
FETCH FIRST 100 ROWS ONLY;
```

הselect מחזיר כמו בפונקציה רשומה של 100 בסיסים עם כמות האספקה הגדולה/קטנה ביותר לפי הממוצע) במקרה שלנו זה מעלה הממוצע לכן הגדרנו את הselect שיחזיר את ה100 הבסיסים עם כמות האספקה הגדולה ביותר).

ניתן לראות שבמקרה הזה, ממוצע כמויות האספקות (TOTAL_SUPPLY) הוא בסדר יורד, כלומר קיבלנו את 100 הבסיסים בעלי האספקה הגדולה ביותר. אפשר לראות שבעמודה הימנית יש את את כל הקיבולות המתאימות לבסיסים.

אחרי ההרצה:

לפני ההרצה:

	BASE_ID	TOTAL_SUPPLY	CAPACITY
1	3598298	2992	9598
2	1453003	2634	7596
3	2006475	2489	4530
4	8005820	2467	2989
5	9859836	2221	8355
6	1685902	2136	5685
7	4213017	2097	2467
8	2245431	2035	2491
9	9488116	2019	4665
10	1227023	1978	4413
11	1840828	1975	3388
12	7985300	1941	6305
13	2492108	1937	5353
14	7028691	1905	9680
15	3027847	1884	6088
16	3345502	1883	6152
17	4996481	1845	2943
18	8623493	1841	4362
19	1783223	1799	9444

	BASE_ID	TOTAL_SUPPLY	CAPACITY
1	3598298	2992	9548
2	1453003	2634	7546
3	2006475	2489	4480
4	8005820	2467	2939
5	9859836	2221	8305
6	1685902	2136	5635
7	4213017	2097	2417
8	2245431	2035	2441
9	9488116	2019	4615
10	1227023	1978	4363
11	1840828	1975	3338
12	7985300	1941	6255
13	2492108	1937	5303
14	7028691	1905	9630
15	3027847	1884	6038
16	3345502	1883	6102
17	4996481	1845	2893
18	8623493	1841	4312
19	1783223	1799	9394

```
The Avrage is: 864.056224899598393574297188755020080321
Base capacities adjyted successfully.
```

😊 הדרן עלך שלב ג