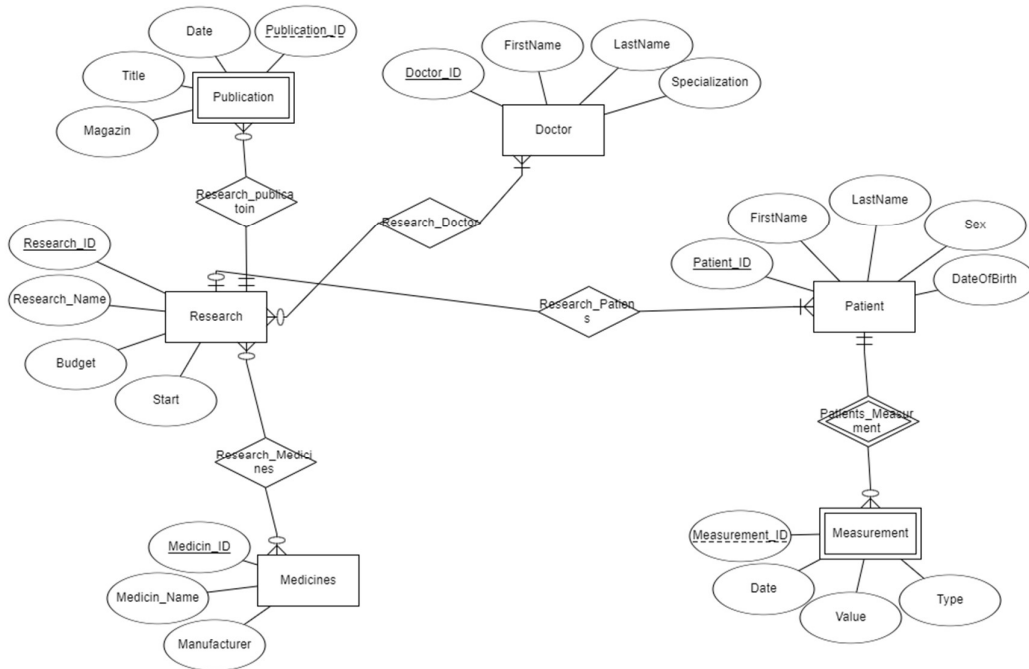


שלב ד אינטגרציה

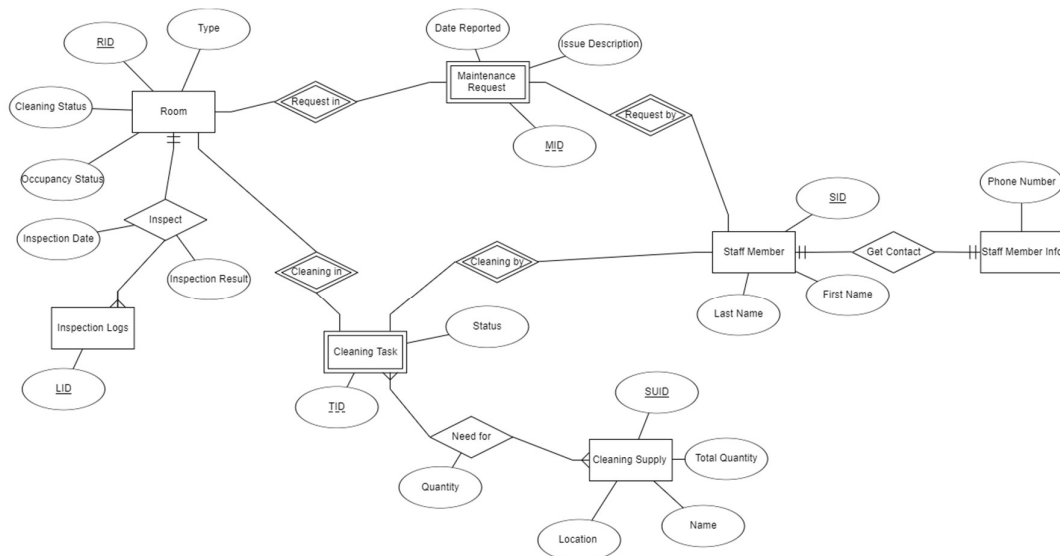
בשלב זה נבצע אינטגרציה של טבלאות SQL של שני האגפים

1. אגף מחקרים
2. אגף ניהול חדרים ונקיון

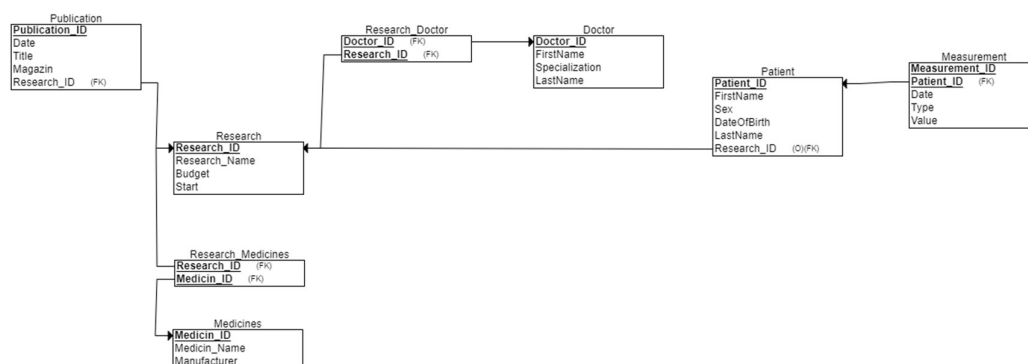
תרשים EDR של אגף מחקרים



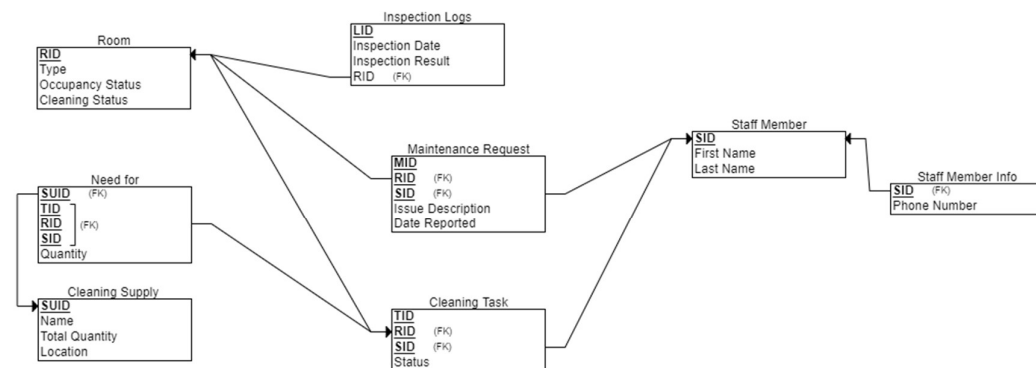
תרשים EDR של אגף ניהול חדרים ונקיון:



תרשים DSD של אגף מחקרים :



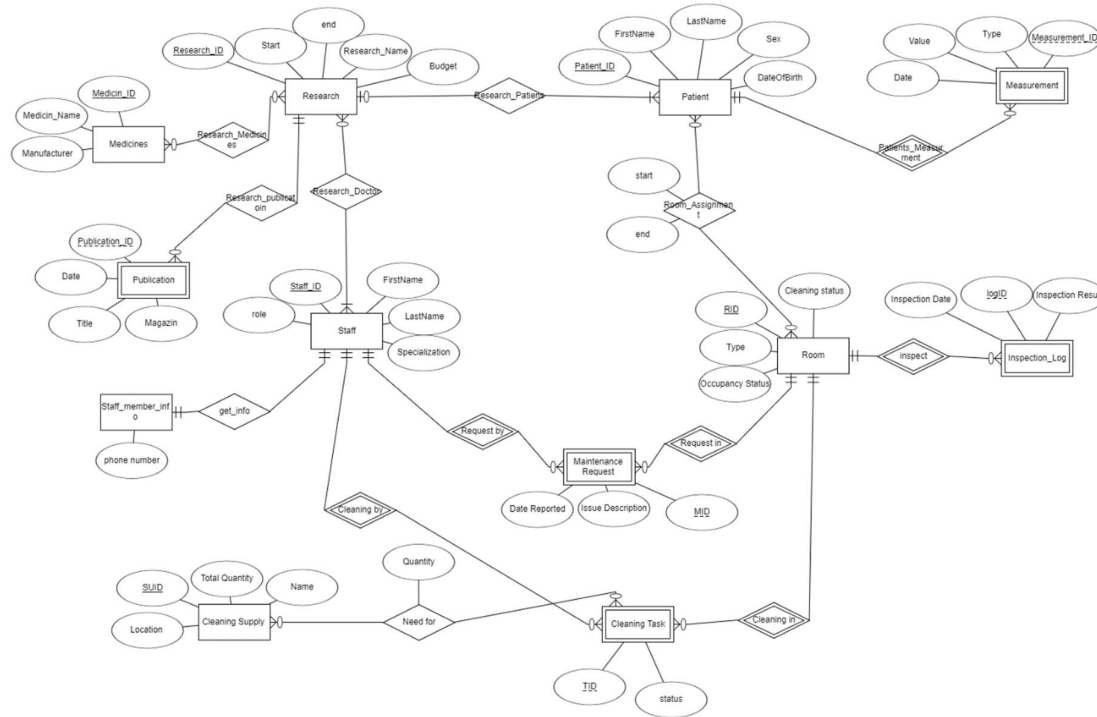
תרשים DSD של אגף נקיון וניהול חדרים:



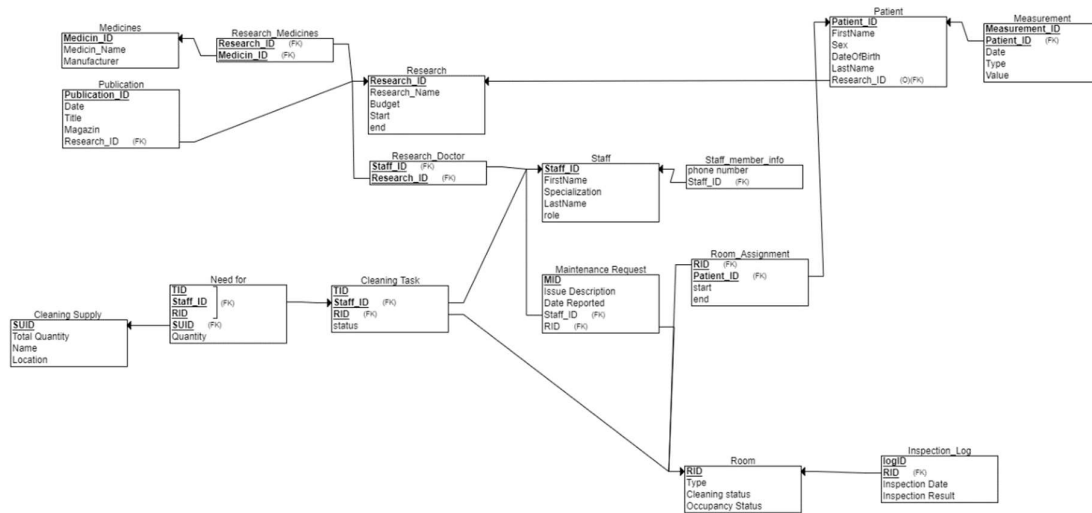
לצורך ביצוע האינטגרציה נבצע כמה שינויים:

1. נאחד את טבלת doctor ואת טבלת staff member לטבלה אחת בשם staff
(כאשר נוסיף עמודה בשם role שבה יהיה כתוב האם חבר הצוות הוא עובד תחזוקה או דוקטור).
2. נבצע קשר חדש בין patient ל room בשם Room_Assignment שבו ישמר מידע על אישפוזי המטופלים.
3. נגדיר את inspection los להיות ישות חלשה לגבי חדר ונוסיף לה את התכונות date ו result
4. נוסיף לטבלת staff_member_info עמודה בשם email

תרשים EDR של האגפים המאוחדים:



תרשים DSD של האגפים המאוחדים:



ניצור את האיחוד בין שני הטבלאות doctor ו staff_memmber
ע"י פקודות alter table וכן נעדכן את כל המצביעים הקיימים בתרשים:

```
--Step 1: Create the new Staff table
CREATE TABLE Staff (
    Staff_ID INT NOT NULL,
    FirstName VARCHAR2(30) NOT NULL,
    LastName VARCHAR2(30) NOT NULL,
    Role VARCHAR2(20) NOT NULL,
    Specialisation VARCHAR2(20),
    PRIMARY KEY (Staff_ID)
);

-- Step 2: Migrate data from Doctor and Staff_Member to Staff
INSERT INTO Staff (Staff_ID, FirstName, LastName, Role, Specialisation)
SELECT Doctor_ID, FirstName, LastName, 'Doctor', Specialisation
FROM Doctor;

INSERT INTO Staff (Staff_ID, FirstName, LastName, Role)
SELECT SID, First_Name, Last_Name, 'Cleaning Staff'
FROM Staff_Member;

-- Step 3: Modify Staff_Member_Info table
ALTER TABLE Staff_Member_Info
RENAME COLUMN SID TO Staff_ID;

ALTER TABLE Staff_Member_Info
ADD Email VARCHAR2(50);

-- Step 5: Modify existing tables to reference Staff instead of Doctor or Staff_Member
ALTER TABLE Research_Doctor
RENAME TO Research_Staff;

ALTER TABLE Research_Staff
RENAME COLUMN Doctor_ID TO Staff_ID;

ALTER TABLE Research_Staff
ADD CONSTRAINT FK_Research_Staff_Staff
FOREIGN KEY (Staff_ID) REFERENCES Staff(Staff_ID);

ALTER TABLE Cleaning_Task
RENAME COLUMN SID TO Staff_ID;

ALTER TABLE Cleaning_Task
ADD CONSTRAINT FK_Cleaning_Task_Staff
FOREIGN KEY (Staff_ID) REFERENCES Staff(Staff_ID);

ALTER TABLE Maintenance_Request
RENAME COLUMN SID TO Staff_ID;

ALTER TABLE Maintenance_Request
ADD CONSTRAINT FK_Maintenance_Request_Staff
FOREIGN KEY (Staff_ID) REFERENCES Staff(Staff_ID);

ALTER TABLE Need_for
RENAME COLUMN SID TO Staff_ID;

ALTER TABLE Need_for
ADD CONSTRAINT FK_Need_for_Staff
FOREIGN KEY (Staff_ID) REFERENCES Staff(Staff_ID);

-- Step 6: Drop old tables
DROP TABLE Doctor;
DROP TABLE Staff_Member;
```

ניצור טבלה חדשה בשם room assignment שבה ישמר המידע על אשפוז החולים:

```
CREATE TABLE Room_Assignment (  
    Assignment_ID INT NOT NULL,  
    Patient_ID INT NOT NULL,  
    RID INT NOT NULL,  
    Admission_Date DATE NOT NULL,  
    Discharge_Date DATE,  
    PRIMARY KEY (Assignment_ID),  
    FOREIGN KEY (Patient_ID) REFERENCES Patient(Patient_ID),  
    FOREIGN KEY (RID) REFERENCES Room(RID)  
);
```

וכן נמלא אותה בנתונים ע"י data generator

ניצור שני מבטים:

מבט 1 : מחלקת מחקרים

```
CREATE VIEW Research_Department_View AS  
SELECT  
    R.Research_ID,  
    R.Research_Name,  
    R.Budget,  
    R.R_Start AS Research_Start_Date,  
    P.Patient_ID,  
    P.FirstName AS Patient_FirstName,  
    P.LastName AS Patient_LastName,  
    P.Sex AS Patient_Sex,  
    P.DateOfBirth AS Patient_DOB,  
    M.Measurement_ID,  
    M.M_Date AS Measurement_Date,  
    M.M_Type AS Measurement_Type,  
    M.M_Value AS Measurement_Value,  
    D.Staff_ID AS Doctor_ID,  
    D.FirstName AS Doctor_FirstName,  
    D.LastName AS Doctor_LastName,  
    D.Specialization AS Doctor_Specialization,  
    SMI.Phone_Number AS Doctor_Phone,  
    SMI.Email AS Doctor_Email,  
    PUB.Publication_ID,  
    PUB.P_Date AS Publication_Date,  
    PUB.Magazin AS Publication_Magazine,  
    RM.Medicin_ID  
FROM Research R  
LEFT JOIN Patient P ON R.Research_ID = P.Research_ID  
LEFT JOIN Measurement M ON P.Patient_ID = M.Patient_ID  
LEFT JOIN Research_Staff RS ON R.Research_ID = RS.Research_ID  
LEFT JOIN Staff D ON RS.Staff_ID = D.Staff_ID  
LEFT JOIN Staff_Member_Info SMI ON D.Staff_ID = SMI.Staff_ID  
LEFT JOIN Publication PUB ON R.Research_ID = PUB.Research_ID  
LEFT JOIN Research_Medicines RM ON R.Research_ID = RM.Research_ID  
WHERE D.Role = 'Doctor';
```

מבט 2 : מחלקת תחזוקה

```
CREATE VIEW Housekeeping_Department_View AS
SELECT
    R.RID,
    R.Type AS Room_Type,
    R.Occupancy_Status,
    R.Cleaning_Status,
    CT.TID AS Cleaning_Task_ID,
    CT.Status AS Cleaning_Task_Status,
    MR.MID AS Maintenance_Request_ID,
    MR.Issue_Description,
    MR.Date_Reported AS Maintenance_Date_Reported,
    IL.LID AS Inspection_Log_ID,
    IL.Inspection_Date,
    IL.Inspection_Result,
    S.Staff_ID,
    S.FirstName AS Staff_FirstName,
    S.LastName AS Staff_LastName,
    SMI.Phone_Number AS Staff_Phone,
    SMI.Email AS Staff_Email,
    CS.SUID AS Cleaning_Supply_ID,
    CS.Name AS Cleaning_Supply_Name,
    CS.Total_Quantity AS Cleaning_Supply_Total_Quantity,
    CS.Location AS Cleaning_Supply_Location,
    NF.Quantity AS Supply_Quantity_Needed
FROM Room R
LEFT JOIN Cleaning_Task CT ON R.RID = CT.RID
LEFT JOIN Maintenance_Request MR ON R.RID = MR.RID
LEFT JOIN Inspection_Logs IL ON R.RID = IL.RID
LEFT JOIN Staff S ON (CT.Staff_ID = S.Staff_ID OR MR.Staff_ID = S.Staff_ID)
LEFT JOIN Staff_Member_Info SMI ON S.Staff_ID = SMI.Staff_ID
LEFT JOIN Need_for NF ON (CT.TID = NF.TID AND CT.RID = NF.RID AND CT.Staff_ID = NF.Staff_ID)
LEFT JOIN Cleaning_Supply CS ON NF.SUID = CS.SUID
WHERE S.Role = 'Cleaning Staff';
```

נבצע שאילתות על המבט למחלקת מחקרים:

שאילה 1 :

מציאת 3 המחקרים עם המספר הגבוהה ביותר של מטופלים

```
SELECT Research_ID, Research_Name, COUNT(DISTINCT Patient_ID) as Patient_Count
FROM Research_Department_View
GROUP BY Research_ID, Research_Name
ORDER BY Patient_Count DESC
FETCH FIRST 3 ROWS ONLY;
```

	RESEARCH_ID	RESEARCH_NAME	PATIENT_COUNT
1	409461	Diuretics in Heart Failure	6
2	695349	Dietary Supplements	5
3	329844	PPIs and Bone Health	5

שאלה 2:

שמות ופרטי הרופאים שלמחקר שהם מבצעים קיים תקציב של מעל 900000 שח

```
SELECT DISTINCT Doctor_ID, Doctor_FirstName, Doctor_LastName, Doctor_Phone, Doctor_Email, Research_Name, Budget
FROM Research_Department_View
WHERE Budget > 900000
ORDER BY Budget DESC;
```

	DOCTOR_ID	DOCTOR_FIRSTNAME	DOCTOR_LASTNAME	DOCTOR_PHONE	DOCTOR_EMAIL	RESEARCH_NAME	BUDGET
1	973619565	Sonny	Stormare	Antipsoriatic Medications	984285
2	867332252	Bridgette	Stevenson	PPIs and Bone Health	951652
3	246916335	Tommy	Biehn	Beta-Blockers	950762
4	740154468	Lorraine	Tolkan	Antiviral Medications	948622
5	204953009	Jerry	Salt	Growth Hormone Therapy	941847
6	681298899	Caroline	Phifer	Beta-Blockers	935391
7	377321271	Delroy	Day	Beta-Blockers	935391
8	302545649	Robbie	Garber	Beta-Blockers	935391
9	962615553	Todd	Cummings	Beta-Blockers	935391
10	412735057	Candice	Rossellini	ARBs and Blood Pressure	927843
11	787562807	Geggy	Smurfit	ARBs and Blood Pressure	927843
12	234307552	Colm	Chao	NSAIDs in Arthritis	904850

שאלות על מבט מחלקת תחזוקה:

שאלה 1:

קבלת כל החדרים שלא נבדקו ב 30 יום האחרונים:

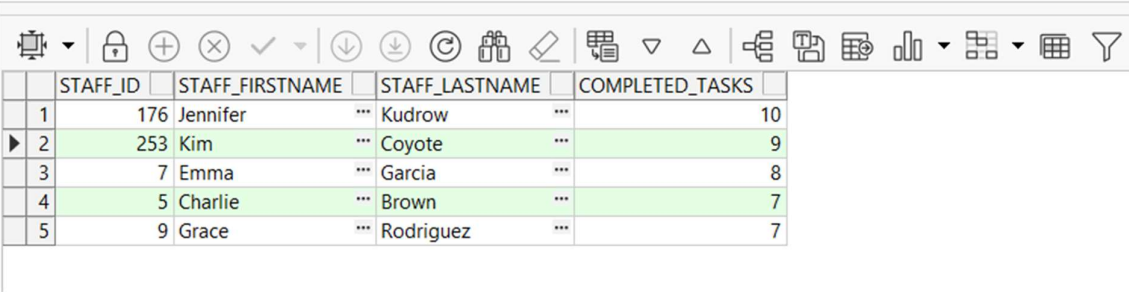
```
SELECT RID, Room_Type, Occupancy_Status, Cleaning_Status, MAX(Inspection_Date) as Last_Inspection_Date
FROM Housekeeping_Department_View
GROUP BY RID, Room_Type, Occupancy_Status, Cleaning_Status
HAVING MAX(Inspection_Date) < SYSDATE - 30 OR MAX(Inspection_Date) IS NULL
ORDER BY Last_Inspection_Date ASC NULLS FIRST;
```

	RID	ROOM_TYPE	OCCUPANCY_STATUS	CLEANING_STATUS	LAST_INSPECTION_DATE
1	20	Single	Occupied	Dirty	...
2	287	Single	Occupied	Dirty	...
3	329	Single	Occupied	Dirty	...
4	239	Single	Occupied	Dirty	...
5	202	Suite	Occupied	Clean	...
6	47	Single	Vacant	In Progress	...
7	341	Double	Vacant	In Progress	...
8	93	Single	Vacant	Clean	...
9	361	Double	Occupied	Clean	...
10	246	Suite	Vacant	In Progress	...
11	108	Single	Occupied	Clean	...
12	307	Suite	Occupied	Dirty	...

שאלתה 2 :

קבלת 5 שמות עובדי תחזוקה שביצעו הכי הרבה פעולות תחזוקה בשנה האחרונה

```
SELECT Staff_ID, Staff_FirstName, Staff_LastName, COUNT(*) as Completed_Tasks
FROM Housekeeping_Department_View
WHERE Cleaning_Task_Status = 'Completed' AND Cleaning_Task_ID IS NOT NULL
      AND TRUNC(SYSDATE) - TRUNC(Inspection_Date) <= 365
GROUP BY Staff_ID, Staff_FirstName, Staff_LastName
ORDER BY Completed_Tasks DESC
FETCH FIRST 5 ROWS ONLY;
```



	STAFF_ID	STAFF_FIRSTNAME	STAFF_LASTNAME	COMPLETED_TASKS
1	176	Jennifer	Kudrow	10
2	253	Kim	Coyote	9
3	7	Emma	Garcia	8
4	5	Charlie	Brown	7
5	9	Grace	Rodriguez	7