/\*Question 1\*/

/\*We only need the Nurse table and to interpret what is in the registered column

to be able to execute against the request\*/

SELECT [EmployeeID]

,[Name]

,[Position]

,[Registered]

,[SSN]

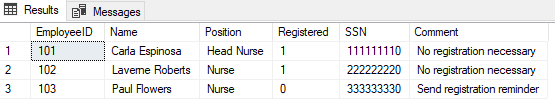
,case

when [Registered] = 1 then 'No registration necessary'

when [Registered] = 0 then 'Send registration reminder'

end as Comment

FROM [ScrubsHospital].[dbo].[Nurse];



/\*Question 2\*/

/\*I inner joined Physician to Department in order to be able to

retrieve the Employee ID that is the head of each department then

pulled the details of each Physician using the Physician table\*/

SELECT [DepartmentID],

[Department].[Name] AS Department,

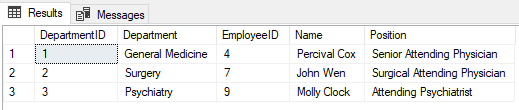
[Physician].[EmployeeID],

[Physician].[Name],

[Physician].[Position]

FROM [ScrubsHospital].[dbo].[Department]

Inner Join [ScrubsHospital].[dbo].[Physician] on [Department].[Head] = [Physician].[EmployeeID];



/\*Question 3\*/

/\*This query returns all patients that utilized examination room A.\*/

SELECT [Patient].[SSN]

,[Patient].[Name]

,[Patient].[Address]

,[Patient].[Phone]

,[Patient].[InsuranceID]

,[Patient].[PCP]

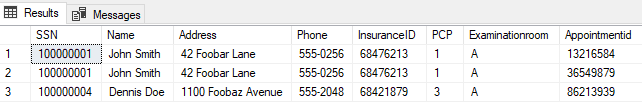
,[Appointment].[Examinationroom]

,[Appointment].[Appointmentid]

FROM [ScrubsHospital].[dbo].[Patient]

Inner join [ScrubsHospital].[dbo].[Appointment] on [Patient].[SSN] = [Appointment].[Patient]

WHERE [ScrubsHospital].[dbo].[Appointment].[Examinationroom] like 'A';



/\*Question 4\*/

/\*I edited the previous query for only B instead of A and removed the examination room as a column

because it is a text field. I also ran it without Distinct to verify that there were duplicate values

then added Distinct to remove those\*/

SELECT DISTINCT

[Patient].[SSN]

,[Patient].[Name]

,[Patient].[Address]

,[Patient].[Phone]

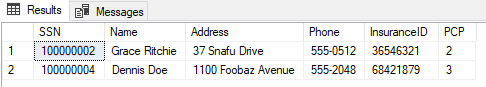
,[Patient].[InsuranceID]

,[Patient].[PCP]

FROM [ScrubsHospital].[dbo].[Patient]

Inner join [ScrubsHospital].[dbo].[Appointment] on [Patient].[SSN] = [Appointment].[Patient]

WHERE [ScrubsHospital].[dbo].[Appointment].[Examinationroom] like 'B';



/\*Question 5\*/

/\*This could also be done as a sub query potentially, but I choose to just use two inner joins and eliminate

the columns that were unnecessary\*/

SELECT DISTINCT

P.[Name] AS Physician\_Name

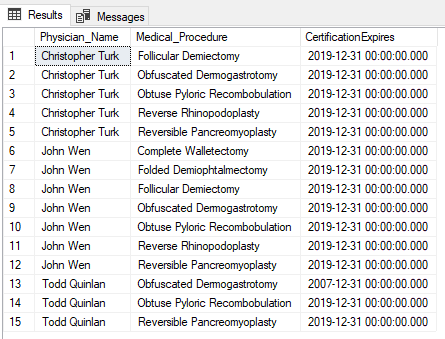
,MP.[NAME] AS Medical\_Procedure

,TI.[CertificationExpires]

FROM [ScrubsHospital].[dbo].[Physician] AS P

INNER JOIN [ScrubsHospital].[dbo].[Trained\_In] AS TI on P.[EmployeeID] = TI.Physician

INNER JOIN [ScrubsHospital].[dbo].[MedProcedures] AS MP on MP.[Code] = TI.Treatment;



/\*Question 6\*/

/\* I used Left Joins to expose where someone might be not affiliated directly with a department

and filtered with null to only get those records with that non affiliation in the final table\*/

SELECT DISTINCT

P.[Name]

,AW.[Department]

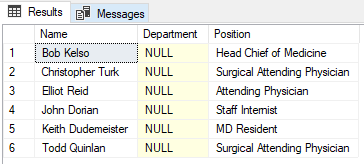
,P.[Position]

FROM [ScrubsHospital].[dbo].[Physician] AS P

LEFT JOIN [ScrubsHospital].[dbo].[Department] AS D on P.[EmployeeID] = D.[Head]

LEFT JOIN [ScrubsHospital].[dbo].[Affiliated\_With] AS AW on AW.[Department] = D.[DepartmentID]

WHERE AW.[Department] is null;



/\*Question 7\*/

/\*Use results from Question 5 as a subquery into this join between Physician and Patient\*/

USE ScrubsHospital;

GO

SELECT

PA.[Name] AS Patient\_Name

,P2.[Name] AS Primary\_Care\_Physician\_Name

FROM [Patient] as PA

INNER JOIN [ScrubsHospital].[dbo].[Physician] AS P2 ON PA.[PCP] = P2.[EmployeeID]

WHERE [PCP] in

(SELECT DISTINCT

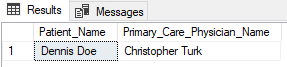
P.[EmployeeID]

FROM [ScrubsHospital].[dbo].[Physician] AS P

INNER JOIN [ScrubsHospital].[dbo].[Trained\_In] AS TI on P.[EmployeeID] = TI.Physician

INNER JOIN [ScrubsHospital].[dbo].[MedProcedures] AS MP on MP.[Code] = TI.Treatment);

GO



/\*Question 8\*/

/\*I used the Appointment table to get both the physician and patient information

then I joined these tables to get more detail. After connecting these tables

I grouped by patient and removed all records that didn't have more than one

doctor\*/

SELECT DISTINCT

PA.[Name] AS Patient\_Name,

COUNT(DISTINCT(P.[Name])) AS Count\_of\_Physicians

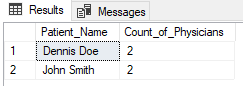
FROM [ScrubsHospital].[dbo].[Appointment] AS A

LEFT JOIN [ScrubsHospital].[dbo].[Patient] AS PA ON A.[Patient] = PA.[SSN]

LEFT JOIN [ScrubsHospital].[dbo].[Physician] AS P ON A.[Physician] = P.[EmployeeID]

GROUP BY PA.[Name]

HAVING COUNT(DISTINCT(P.[Name]))>1;



/\*Question 9\*/

/\*I started by checking dates and filtering for that first then after I got the right appointments

I started to add the query from question 8. I did notice a record that had a start date of 4/27

and an end date of 4/21, which is why I filtered on start date and not end date. I am assuming this

record is erroneous so I left it out. I used 4/25 as my end date because it is starting at 00:00:00 seconds

when interpreting, so it did not pull any actual 4/25 appointments into the query results\*/

SELECT DISTINCT

PA.[Name] AS Patient\_Name

,P.[Name] AS Physician\_Name

,N.[Name] AS Nurse\_Name

,A.[Start]

,A.[End]

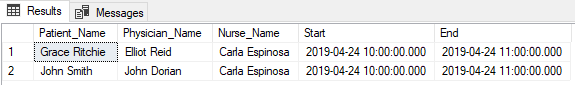
FROM [ScrubsHospital].[dbo].[Appointment] AS A

LEFT JOIN [ScrubsHospital].[dbo].[Patient] AS PA ON A.[Patient] = PA.[SSN]

LEFT JOIN [ScrubsHospital].[dbo].[Physician] AS P ON A.[Physician] = P.[EmployeeID]

LEFT JOIN [ScrubsHospital].[dbo].[Nurse] AS N ON A.[PrepNurse] = N.[EmployeeID]

WHERE A.[Start] BETWEEN '20190421' AND '20190425';



/\*Question 10\*/

/\*I connected the Patient, Prescribes and Medication table where Appointment was null.

I initially started with using the Appointment table and the Patient table, but I was only

retrieving records with an appointment and could not find patients that did not have an

appointment\*/

SELECT

PA.[Name] AS Patient\_Name

,PRS.[Medication]

,MED.[Name] AS Drug\_Name

,MED.[Brand]

,PRS.[Dose]

FROM [ScrubsHospital].[dbo].[Patient] AS PA

RIGHT JOIN [ScrubsHospital].[dbo].[Prescribes] AS PRS ON PRS.[Patient] = PA.[SSN]

INNER JOIN [ScrubsHospital].[dbo].[Medication] AS MED ON MED.[Code]=PRS.[Medication]

WHERE PRS.[Appointment] is null;



/\*Question 11\*/

/\*I first removed all records that were available to get to the subset of data that needed to be rolled up.

I then started to remove the columns we did not need in our query and did a check to see if there were any repeating

block floors and block codes, which there was not. I rolled everything up by using group by and then sorted using order by to get to the final table.\*/

SELECT

R.[BlockFloor] AS [Floor]

,R.[BlockCode] AS [Block]

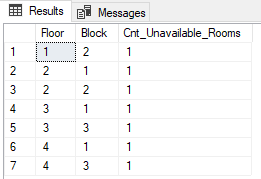
,COUNT(R.[Unavailable]) AS Cnt\_Unavailable\_Rooms

FROM [ScrubsHospital].[dbo].[Room] AS R

WHERE R.[Unavailable] = 1

GROUP BY R.[BlockFloor], R.[BlockCode]

ORDER BY R.[BlockFloor], R.[BlockCode];



/\*Question 12\*/

/\*Each Floor has 9 total rooms. The only floor that has 8 rooms available is floor 1

and everything else is less than that. I did an initial view on all floors and then filtered

in the having clause for floor 1 when I verified it was the floor with the minimum amount of

rooms unavailable\*/

SELECT

R.[BlockFloor] AS [Floor]

,"Cnt\_Avail" =

case when R.[Unavailable]=0 then count(R.[Unavailable])

end

,"Cnt\_Unvail" =

case when R.[Unavailable]=1 then count(R.[Unavailable])

end

FROM [ScrubsHospital].[dbo].[Room] AS R

GROUP BY R.[BlockFloor], R.[Unavailable]

HAVING R.[BlockFloor]=1

ORDER BY R.[BlockFloor];

