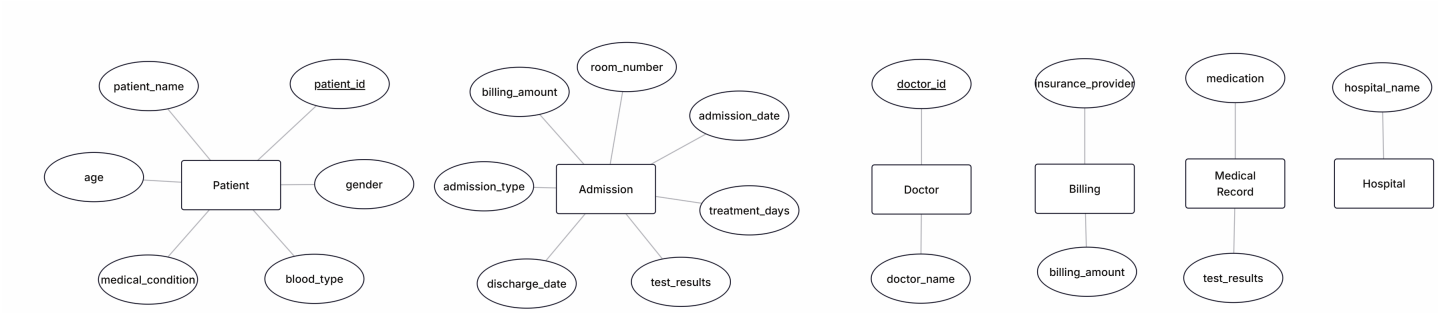
Healthcare SQL Data Analysis

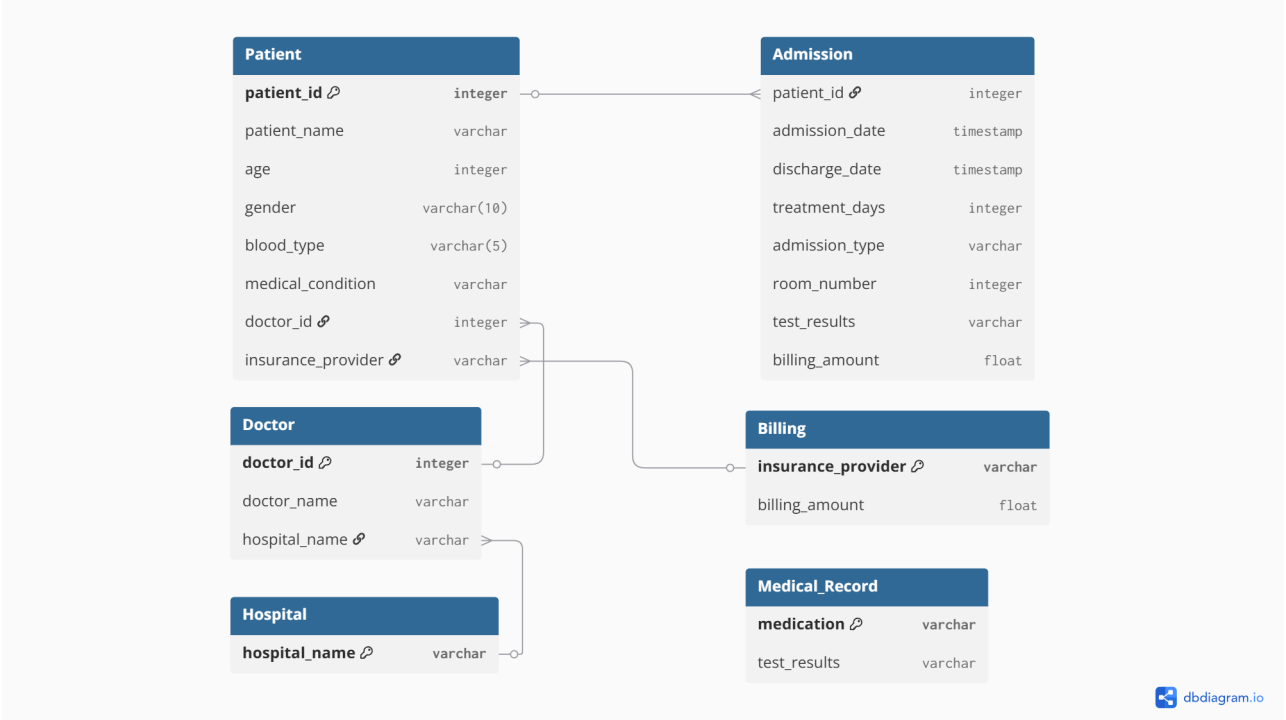
Contents

Analysis Preparation

1.1 Diagrams

In order to work with SQL, relational diagrams must be made to ensure clarity with the attributes to see what tables must be created and what relations a table can have with another. The following tables were created:





With this visualization, 6 tables were created for organization and to create relationships one another.

SQL Tables

2.1 Hospital and Doctor Tables

```sql
CREATE TABLE Hospital (
    hospital_name VARCHAR(150) PRIMARY KEY
);

CREATE TABLE Doctor (
    doctor_id INT PRIMARY KEY,
    doctor_name VARCHAR(100),
    hospital_name VARCHAR(150),
    CONSTRAINT FK_Doctor_Hospital FOREIGN KEY (hospital_name)
   REFERENCES Hospital(hospital_name)
);
```

2.2 Billing Table

```sql
CREATE TABLE Billing (
    insurance_provider VARCHAR(100) PRIMARY KEY,
    billing_amount DECIMAL(10,2)
);
```

2.3 Patient Table

```sql
CREATE TABLE Patient (
    patient_id INT PRIMARY KEY,
    patient_name VARCHAR(100),
    age INT,
    gender VARCHAR(10),
    blood_type CHAR(3),
    medical_condition VARCHAR(100),
    doctor_id INT,
    insurance_provider VARCHAR(100),
    CONSTRAINT FK_Patient_Doctor FOREIGN KEY (doctor_id) REFERENCES
   Doctor(doctor_id),
    CONSTRAINT FK_Patient_Billing FOREIGN KEY (insurance_provider)
   REFERENCES Billing(insurance_provider)
```

```
);
```

2.4 Admission Table

```
CREATE TABLE Admission (
    admission_id INT IDENTITY(1,1) PRIMARY KEY,
    patient_id INT,
    admission_date DATE,
    discharge_date DATE,
    treatment_days INT,
    admission_type VARCHAR(20),
    room_number INT,
    test_results VARCHAR(20),
    billing_amount DECIMAL(10,2),
    CONSTRAINT FK_Admission_Patient FOREIGN KEY (patient_id)
   REFERENCES Patient(patient_id)
);
```

2.5 Medical Record Table

```
CREATE TABLE Medical_Record (
    medication VARCHAR(100) PRIMARY KEY,
    test_results VARCHAR(20)
);
```

SQL Queries and Analysis

3.1 Patient Basic Statistics and Demographics

3.1.1 Total Patients in the Database

```
SELECT COUNT(*) AS total_patients FROM Patient;
```

Findings: Displays the total number of patients in the dataset, in this case there are 55,392 patients in total.

3.1.2 Patient Count by Gender

```
SELECT gender, COUNT(*) AS gender_count
FROM Patient
GROUP BY gender;
```

Findings: Shows how many patients are either male or female in the dataset, there are 27,526 male patients and 27,666 female patients.

3.1.3 Distinct Medical Conditions among Patients

```
SELECT DISTINCT medical_condition
FROM Patient;
```

Findings: There are 5 distinct medical conditions throughout this dataset:

Condition
Arthritis
Obesity
Asthma
Hypertension
Cancer
Diabetes

3.1.4 Minimum and Maximum Age of Patients

```
SELECT MIN(age) AS min_age, MAX(age) AS max_age
FROM Patient;
```

Findings: The youngest patient in this dataset is 13 years old and the oldest is 89 years old.

3.1.5 Oldest to Youngest Patients

```
SELECT patient_name,
       age
FROM Patient
ORDER BY age DESC;
```

Findings: Displays all patients' names and age in descending order.

3.1.6 Top 10 Oldest Patients with Conditions

```
SELECT TOP 10 patient_name,
              medical_condition,
              age
FROM Patient
ORDER BY age DESC;
```

Findings: The ages in this query are 89 year old (as they are the oldest), and the conditions in this query contains:

Patient Count	Condition
2	Arthritis
1	Obesity
1	Asthma
4	Hypertension
1	Cancer
1	Diabetes

There is likely to be more senior citizens who have hypertension than of the other medical conditions.

3.1.7 Top 10 Youngest Patients with Conditions

```
SELECT TOP 10 patient_name,
              medical_condition,
              age
FROM Patient
ORDER BY age;
```

Findings: This query contains patients that are 13 years old, as they are the youngest, the conditions in this query contain:

Patient Count	Condition
2	Asthma
5	Obesity
2	Arthritis
1	Cancer

Under the assumption that obesity is more common than the other conditions present, it would be imperative to promote healthy decisions to reduce their weight and remain healthy.

3.1.8 Patient Blood Types Distribution

```
SELECT blood_type,
        COUNT(*) as Patient_Count
FROM Patient
GROUP BY blood_type
ORDER BY Patient_Count DESC;
```

Findings: There is a uniform distribution, meaning the counts of patients per blood type are almost equal to each other.

Blood Type	Patient Count
A-	6953
A+	6939
AB+	6937
B+	6936
B-	6933
AB-	6929
O+	6905
O-	6860

3.1.9 Average Age by Gender

```
SELECT AVG(age) as AverageAge, gender
FROM Patient
GROUP BY gender
ORDER BY AverageAge DESC;
```

Findings: Both male and female average age is 51.

3.2 Age Groups and Condition Analysis

3.2.1 Medical Conditions for Patients Over 65

```
SELECT medical_condition,
       COUNT(*) AS number_of_patients_over_65
FROM Patient
WHERE age > 65
GROUP BY medical_condition;
```

Findings: Similar to blood types, the distribution of patient counts are close to each other:

Medical Condition	Number of Patients Over 65
Arthritis	2745
Obesity	2606
Asthma	2734
Hypertension	2739
Cancer	2705
Diabetes	2689

Interestingly enough, more patients have arthritis, asthma and hypertension then the other 3 conditions which can be an interpretation of due to older age they will have these conditions or genetics in the case of asthma or the other ones (aside from obesity)

3.2.2 Distribution of Medical Conditions Across Age Groups

```sql
SELECT
    CASE
        WHEN age BETWEEN 10 AND 20 THEN '10-20'
        WHEN age BETWEEN 21 AND 30 THEN '21-30'
        WHEN age BETWEEN 31 AND 40 THEN '31-40'
        WHEN age BETWEEN 41 AND 50 THEN '41-50'
        WHEN age BETWEEN 51 AND 60 THEN '51-60'
        WHEN age BETWEEN 61 AND 70 THEN '61-70'
        WHEN age BETWEEN 71 AND 80 THEN '71-80'
        ELSE '81+'
    END AS age_group,
    medical_condition,
    COUNT(*) AS total_patients
FROM Patient
GROUP BY
    CASE
        WHEN age BETWEEN 10 AND 20 THEN '10-20'
        WHEN age BETWEEN 21 AND 30 THEN '21-30'
        WHEN age BETWEEN 31 AND 40 THEN '31-40'
        WHEN age BETWEEN 41 AND 50 THEN '41-50'
        WHEN age BETWEEN 51 AND 60 THEN '51-60'
        WHEN age BETWEEN 61 AND 70 THEN '61-70'
        WHEN age BETWEEN 71 AND 80 THEN '71-80'
        ELSE '81+'
    END,
    medical_condition
ORDER BY age_group, total_patients DESC;
```

Findings: There are multiple groupings of this query, so it will be separated into different tables to illustrate counts more effectively:

(10-20) Condition	Patient Count
Arthritis	427
Cancer	416
Asthma	407
Obesity	403
Diabetes	399
Hypertension	385

(21-30) Condition	Patient Count
Obesity	1389
Cancer	1357
Asthma	1338
Hypertension	1336
Diabetes	1325
Arthritis	1300

(31-40) Condition	Patient Count
Arthritis	1399
Asthma	1355
Diabetes	1354
Hypertension	1347
Cancer	1334
Obesity	1318

(41-50) Condition	Patient Count
Diabetes	1413
Arthritis	1379
Obesity	1378
Hypertension	1353
Asthma	1337
Cancer	1330

(51-60) Condition	Patient Count
Obesity	1418
Arthritis	1401
Cancer	1390
Hypertension	1381
Diabetes	1381
Asthma	1309

(61-70) Condition	Patient Count
Diabetes	1456
Asthma	1377
Obesity	1363
Hypertension	1349
Cancer	1342
Arthritis	1330

(71-80) Condition	Patient Count
Hypertension	1394
Arthritis	1374
Asthma	1369
Cancer	1362
Diabetes	1318
Obesity	1275

(81+) Condition	Patient Count
Arthritis	687
Hypertension	679
Cancer	677
Asthma	675
Obesity	668
Diabetes	638

The tables showcase the distributions of patients with medical condition based off their age group

3.3 Hospital and Doctor Analytics

3.3.1 Patients per Hospital

```
SELECT
  d.hospital_name,
  COUNT(p.patient_id) AS total_patients
FROM Patient p
JOIN Doctor d ON p.doctor_id = d.doctor_id
GROUP BY d.hospital_name
ORDER BY total_patients DESC;
```

Findings: The highest patient in a hospital are 30-37, and the lowest being 1. There are 33,602 rows of distinct hospitals where patient(s) are located in so if there is crowding or issues there are various hospitals a patient can be administered to.

3.3.2 Top 10 Doctors with Most Patients

```
SELECT TOP 10
  d.doctor_name,
  COUNT(p.patient_id) as total_patients
FROM Patient p
JOIN Doctor d on p.doctor_id = d.doctor_id
GROUP BY d.doctor_name
ORDER BY total_patients DESC;
```

Findings: This shows that Dr Michael Smith has a lot of patients compared to the other ones in the top 10 patients. It could mean he is overworked or they are patients that have similiar needs.

Doctor Name	Total Patients
Michael Smith	27
John Smith	22
Robert Smith	22
Michael Johnson	20
David Smith	19
James Smith	19
Robert Johnson	19
Michael Williams	18
John Johnson	17
Christopher Smith	17

3.3.3 Most Common Conditions per Hospital

```sql
SELECT
  d.hospital_name,
  p.medical_condition,
  COUNT(*) as case_count
FROM Patient p
JOIN Doctor d ON p.doctor_id = d.doctor_id
GROUP BY d.hospital_name, p.medical_condition
ORDER BY d.hospital_name, case_count DESC;
```

Findings: This query showed 44,114 rows of information showing how many cases per medical condition in a hospital. For most of these hospitals it is 1 case of a medical condition in a hospital followed by either 2+ conditions in the same hospital.

3.4 Treatment and Billing Analytics

3.4.1 Average Treatment Days by Condition

```sql
SELECT
  p.medical_condition,
  AVG(a.treatment_days) AS average_days
FROM Patient p
JOIN Admission a ON p.patient_id = a.patient_id
GROUP BY p.medical_condition
ORDER BY average_days DESC;
```

Findings: The average days on all the condition are 15 days.

3.4.2 Average Treatment Days by Admission Type

```sql
SELECT
  admission_type,
  AVG(treatment_days) AS avg_days
FROM Admission
GROUP BY admission_type
ORDER BY avg_days DESC;
```

Findings: Same thing as by condition, 15 days.

3.4.3 Total Billing by Insurance Provider

```sql
SELECT
  b.insurance_provider,
  SUM(a.billing_amount) AS total_bill
FROM Billing b
JOIN Patient p ON b.insurance_provider = p.insurance_provider
JOIN Admission a ON p.patient_id = a.patient_id
GROUP BY b.insurance_provider
ORDER BY total_bill DESC;
```

Findings: The total billings of all insurance providers are enormous as they are covering millions of patients worldwide, but in this dataset it is still expensive. It also can be an interpretation of what the "popular" insurance provider is, in this case it could be Cigna or Medicare as there is more in their billings than Aetna.

Insurance Provider	Total Bill
Cigna	$287,151,707.11
Medicare	$285,731,211.44
Blue Cross	$283,262,241.74
UnitedHealthcare	$282,459,703.05
Aetna	$278,881,103.14

3.4.4 Total Billing by Blood Type

```sql
SELECT
  p.blood_type,
  SUM(a.billing_amount) AS total_bill
FROM Patient p
JOIN Admission a ON p.patient_id = a.patient_id
GROUP BY p.blood_type
ORDER BY total_bill DESC;
```

Findings: The billings per blood type are close to each other however, the difference between A+ and O+ are more than any of the other blood types.

Blood Type	Total Bill
A+	$178,528,820.70
AB-	$178,461,573.49
A-	$178,378,634.07
O-	$177,408,818.09
B-	$177,245,305.20
B+	$176,612,794.15
AB+	$176,193,267.88
O+	$174,656,752.90

3.4.5 Average Billing for Patients with Long Treatments (Over 15 Days)

```sql
WITH LongTreatments AS (
    SELECT
        p.patient_id,
        a.treatment_days,
        a.billing_amount
    FROM Patient p
    JOIN Admission a ON p.patient_id = a.patient_id
    WHERE a.treatment_days >= 15
)
SELECT
    CAST(ROUND(AVG(CAST(billing_amount AS DECIMAL(10,2))), 2) AS
   DECIMAL(10,2)) AS avg_billing_for_long_treatments
FROM LongTreatments;
```

Findings: The result came to be about $25,516.66. What this value mean is that for patients that stay more than 15 days, their bill will be very expensive on average across the hospitals.

3.4.6 Total Billing per Insurance Provider by Blood Type

```sql
WITH BillingDetails AS (
  SELECT
    p.blood_type,
    p.insurance_provider,
    a.billing_amount
  FROM Patient p
  JOIN Admission a ON p.patient_id = a.patient_id
)
SELECT
  insurance_provider,
  blood_type,
  ROUND(SUM(billing_amount), 2) AS total_billed
FROM BillingDetails
GROUP BY insurance_provider, blood_type
ORDER BY insurance_provider, total_billed DESC;
```

Findings: From these 5 tables, Aetna and Blue Cross are billed less than those of Cigna, UHC, or Medicare. The interpretation of this could be that the two insurances are not popular or they have a different terms and conditions in comparison to the other 3.

(Aetna) Blood Type	Total Billed
O+	$35,798,589.47
B-	$35,534,544.50
A-	$35,130,784.62
AB+	$34,751,658.48
B+	$34,694,543.94
O-	$34,552,323.70
AB-	$34,527,858.98
A+	$33,890,799.45

(Blue Cross) Blood Type	Total Billed
A-	$36,681,066.34
A+	$36,173,473.27
AB-	$35,888,516.89
B+	$35,672,724.58
B-	$35,603,317.64
O+	$34,755,941.34
O-	$34,607,417.77
AB+	$33,879,783.91

(Cigna) Blood Type	Total Billed
A-	$37,285,083.53
A+	$36,973,866.68
AB-	$36,778,734.77
O+	$35,776,180.21
O-	$35,637,045.29
B+	$35,332,546.10
B-	$34,909,342.90
AB+	$34,458,907.63

(Medicare) Blood Type	Total Billed
O-	$37,682,980.51
B-	$37,654,145.37
AB-	$35,798,118.16
AB+	$35,684,926.85
A+	$35,630,773.46
A-	$35,079,258.06
B+	$34,683,490.90
O+	$33,517,518.13

(UnitedHealthcare) Blood Type	Total Billed
AB+	$37,417,991.01
B+	$36,229,488.63
A+	$35,859,907.84
AB-	$35,468,344.69
O-	$34,929,050.82
O+	$34,808,523.75
A-	$34,202,441.52
B-	$33,543,954.79

3.4.7 Top 3 Admission Types by Total Billing for Blood Type 'O'

```sql
WITH BloodOAdmissions AS (
  SELECT
    a.admission_type,
    a.billing_amount
  FROM Patient p
  JOIN Admission a ON p.patient_id = a.patient_id
  WHERE p.blood_type LIKE 'O%'
),
BillingType AS (
SELECT
  admission_type,
  SUM(billing_amount) AS total_billed
FROM BloodOAdmissions
GROUP BY admission_type
)
SELECT
  admission_type,
  ROUND(total_billed, 2) AS total_billed
FROM BillingType
ORDER BY total_billed DESC;
```

Findings: For this particular blood type, more people must have been admitted for electives than emergency as there is a big difference between elective and the other 2 admission types.

Admission Type	Total Billed
Elective	$121,040,177.10
Urgent	$116,159,173.40
Emergency	$114,866,220.49

3.4.8 Total Billing per Patient Stay By Admission Type

```sql
WITH StayClassification AS (
  SELECT
    a.admission_type,
    a.billing_amount,
    CASE
    WHEN a.treatment_days <= 3 THEN 'Short Stay'
    WHEN a.treatment_days <= 5 THEN 'Medium Stay'
    ELSE 'Long Stay'
    END AS stay_category
  FROM Admission a
),
BillingSummary AS (
  SELECT
    stay_category,
    admission_type,
    SUM(billing_amount) AS total_billed
  FROM StayClassification
  GROUP BY stay_category, admission_type
)
SELECT
  stay_category,
  admission_type,
  ROUND(total_billed, 2) AS total_billed
FROM BillingSummary
ORDER BY stay_category, total_billed;
```

Findings: Usually a short stay is considered 72 hours, a medium ~5 days. From that, there are more bills for short stays vs a medium stay. A longer stay would create more bills for all patients, but it is interesting to note the difference for a short vs medium stay.

Admission Type	Long Stay	Medium Stay	Short Stay
Elective	$398,356,068.77	$31,717,154.19	$47,558,646.84
Urgent	$392,766,962.78	$32,882,653.25	$48,380,761.00
Emergency	$390,819,336.18	$30,517,871.23	$44,486,512.24

3.5 Advanced Doctor and Patient Billing Rankings

3.5.1 Doctor's Total Number of Patients and Average Treatment Duration

```sql
WITH DoctorAdmissions AS (
  SELECT
    d.doctor_id,
    d.doctor_name,
    a.treatment_days
  FROM Doctor d
  JOIN Patient p ON d.doctor_id = p.doctor_id
  JOIN Admission a ON p.patient_id = a.patient_id
```

```
),
DoctorStats AS (
  SELECT
    doctor_id,
    doctor_name,
    COUNT(*) AS total_patients,
    ROUND(AVG(treatment_days), 2) AS avg_treatment_days
  FROM DoctorAdmissions
  GROUP BY doctor_id, doctor_name
)
SELECT
  doctor_name,
  total_patients,
  avg_treatment_days
FROM DoctorStats
ORDER BY total_patients DESC;
```

Findings: This query's results showed a long list of rows displaying patients per doctor and average treatment duration for all patients. Of course with the doctors handling the most patients, their average treatment days ranges from 13 to 18. However interestingly, there are doctors who do not have more than 1 patient and take much longer than the average time for doctors handling more patients. A further question of what these patients are admitted to as or the doctor's competency.

3.5.2 Billing Per Month for the Current Year

```
WITH ThisYearAdmissions AS (
  SELECT
    MONTH(a.admission_date) AS admission_month,
    a.billing_amount
  FROM Admission a
),
MonthlyBilling AS (
  SELECT
    admission_month,
    SUM(billing_amount) AS total_billed
  FROM ThisYearAdmissions
  GROUP BY admission_month
)
SELECT
  DATENAME(MONTH, DATEFROMPARTS(YEAR(GETDATE()), admission_month,
   1)) AS month_name,
  ROUND(SUM(total_billed), 2) AS total_billed
FROM MonthlyBilling
GROUP BY admission_month
ORDER BY admission_month;
```

Findings: Interestingly enough, the lowest month of total bills was in February while the highest was in July. Upon reading this, the increase of bills could be a factor of events that had happened in the recording of this dataset or due to seasonal diseases such as the flu, the common cold, or heat stroke.

Month Name	Total Billed
January	$120,531,554.15
February	$107,902,173.41
March	$118,566,596.57
April	$114,564,089.65
May	$116,450,604.18
June	$121,901,639.46
July	$122,757,107.23
August	$122,665,151.22
September	$114,745,118.84
October	$121,594,245.26
November	$116,960,784.74
December	$118,846,901.77

3.5.3 Rank Doctors by Total Billing and Compare to Average

```sql
WITH DoctorBilling AS (
  SELECT
    d.doctor_id,
    d.doctor_name,
    SUM(a.billing_amount) AS total_billed
  FROM Doctor d
  JOIN Patient p ON d.doctor_id = p.doctor_id
  JOIN Admission a ON p.patient_id = a.patient_id
  GROUP BY d.doctor_id, d.doctor_name
),
RankedBilling AS (
  SELECT
    doctor_id,
    doctor_name,
    total_billed,
    RANK() OVER (ORDER BY total_billed DESC) AS billing_rank
  FROM DoctorBilling
)
SELECT
  doctor_name,
  total_billed,
  billing_rank,
  ROUND(total_billed - (
      SELECT AVG(total_billed) FROM DoctorBilling), 2) AS
   billing_vs_avg
FROM RankedBilling
ORDER BY billing_rank;
```

Findings: In this query, there are ranked doctors of who has the highest billing and another column computing the average billing. The difference between the average and total is not that far different until down the rows of data where the average is less than the total. However the difference for total and average for majority of the rows are ∼$30,000

3.5.4 Rank Patients' Total Billing Within Their Blood Type

```sql
WITH PatientBilling AS (
  SELECT
    p.patient_id,
    p.patient_name,
    p.blood_type,
    SUM(a.billing_amount) AS total_billed
  FROM Patient p
  JOIN Admission a ON p.patient_id = a.patient_id
  GROUP BY p.patient_id, p.patient_name, p.blood_type
),
RankedBloodType AS (
  SELECT
    patient_id,
    patient_name,
    blood_type,
    total_billed,
    RANK() OVER (
      PARTITION BY blood_type
      ORDER BY total_billed DESC
      ) AS blood_type_rank
  FROM PatientBilling
)
SELECT
  blood_type,
  patient_name,
  total_billed,
  blood_type_rank
FROM RankedBloodType
ORDER BY blood_type, blood_type_rank;
```

Findings: The results of this query shows separated blood types with their billings ranked. Further analysis can be done to discover the average between all blood types but skimming upon the results, the billings are similar amongst all of them.

3.5.5 Rank Patients by Average Treatment Days and Total Billing

```sql
WITH BillingStats AS (
  SELECT
    p.patient_id,
    p.patient_name,
    SUM(a.billing_amount) AS total_billed
  FROM Patient p
  JOIN Admission a ON p.patient_id = a.patient_id
  GROUP BY p.patient_id, p.patient_name
),
TreatmentStats AS (
  SELECT
    p.patient_id,
    ROUND(AVG(a.treatment_days), 2) AS avg_treatment_days
  FROM Patient p
  JOIN Admission a ON p.patient_id = a.patient_id
  GROUP BY p.patient_id
```

```
),
RankedPatients AS (
  SELECT
    bs.patient_id,
    bs.patient_name,
    bs.total_billed,
    ts.avg_treatment_days,
    RANK() OVER (ORDER BY bs.total_billed DESC) AS billing_rank
  FROM BillingStats AS bs
  JOIN TreatmentStats ts ON bs.patient_id = ts.patient_id
)
SELECT
  patient_name,
  total_billed,
  avg_treatment_days,
  billing_rank
FROM RankedPatients
ORDER BY billing_rank;
```

Findings: Upon skimming through the results of the query, some patients had an expensive billing for being there for 1 to 26 days or such and such. Depending on their condition or admission type, the billing would vary between patients as some had a low billing than others even with the amount of days they had stayed.