**#Patient & Admission Insights**

#1. Find the total number of emergency admissions in the last 6 months.

select count(\*) as emergency\_admissions\_in\_last\_6\_months

from `nodal-descent-466716-a7.My\_data.dataset`

where EMERGENCY\_OR\_SCHEDULED ='Emergency'

and Date\_of\_Admission>= date\_sub( current\_date(), interval 6 month);

#2. List patients with stay durations over 10 days and medical condition is cancer.

select name\_of\_patient from `nodal-descent-466716-a7.My\_data.dataset`

where Stay\_Days>10 and Medical\_Condition = 'Cancer';

#3. Show monthly count of admissions by Admission Type (Emergency vs Scheduled).

SELECT count(`Months\_of\_admission`),

`Admission Type`

FROM `nodal-descent-466716-a7.My\_data.dataset`

WHERE

`Admission Type` in ('Emergency','Scheduled')

group by `Admission Type`;

**#Billing & Insurance Analytics**

#1. Compare total billing amounts across different insurance providers.

SELECT `Insurance Provider`, sum(Billing\_Amount) from `nodal-descent-466716-a7.My\_data.dataset` group by `Insurance Provider`;

#2. Find the average billing amount for patients grouped by Risk Level.

SELECT avg(Billing\_Amount), `Risk\_Level` from `nodal-descent-466716-a7.My\_data.dataset` group by `Risk\_Level`;

#3. Identify doctors with the highest average billing per patient.

SELECT Avg(`Billing\_Amount`), `Doctor` from `nodal-descent-466716-a7.My\_data.dataset` group by`Name\_of\_patient`, `Doctor`;

#4. Which hospital has generated the highest revenue this year?

SELECT HOSPITAL, MAX(Billing\_Amount) FROM `nodal-descent-466716-a7.My\_data.dataset`

GROUP BY Hospital

LIMIT 1;

**#Hospital Operations**

#1. Find room numbers that consistently host cancer patients.

SELECT

Room\_Number

FROM

`nodal-descent-466716-a7.My\_data.dataset`

where Medical\_Condition= 'Cancer';

#2. Identify patients who extended their stay.

select Name\_of\_patient from `nodal-descent-466716-a7.My\_data.dataset`

where `Extend\_Stay` = 'EXTEND STAY';

#3. Calculate average stay days for each hospital.

SELECT

avg(Stay\_Days), `Hospital`

FROM

`nodal-descent-466716-a7.My\_data.dataset`

Group BY Hospital;

#4. Group patients by doctor and find average patient age for each.

SELECT

count(Name\_of\_patient) as count\_of\_patients, Doctor, avg(`Age\_of\_patient`) as average\_age

FROM

`nodal-descent-466716-a7.My\_data.dataset`

Group BY Doctor;

**#Diagnostics & Risk Evaluation**

#1. Get patients whose test results indicate high risk but were scheduled instead of emergency.

SELECT

`Name\_of\_patient`,

FROM

`nodal-descent-466716-a7.My\_data.dataset`

WHERE Risk\_Level= 'High risk' and EMERGENCY\_OR\_SCHEDULED= 'Scheduled';

#2. List all patients with abnormal test results who were discharged within 3 days.

SELECT

`Name\_of\_patient`,

FROM

`nodal-descent-466716-a7.My\_data.dataset`

WHERE Test\_Results='Abnormal' and Stay\_Days<=3;

#3. Rank doctors based on the number of critical cases handled.

SELECT

distinct(Doctor), COUNT(\*) AS critical\_case\_count

FROM

`nodal-descent-466716-a7.My\_data.dataset`

WHERE Medical\_Condition= 'Cancer'

group by Doctor

order by critical\_case\_count DESC;

#4. Find patterns between Medical Conditions and Risk Level.

SELECT

Medical\_Condition,

Risk\_Level

FROM

`nodal-descent-466716-a7.My\_data.dataset`

GROUP BY

Medical\_Condition,

Risk\_Level

ORDER BY

Medical\_Condition, Risk\_Level;

**#Advanced SQL Concepts**

#1. Create a window function to compare billing amount per patient against average hospital billing.

SELECT

Name\_of\_patient

Hospital,

Billing\_Amount,

AVG(Billing\_Amount) OVER (PARTITION BY Hospital) AS Average\_Hospital\_Billing,

ROUND(Billing\_Amount - AVG(Billing\_Amount) OVER (PARTITION BY Hospital), 2) AS Billing\_Difference

FROM

`nodal-descent-466716-a7.My\_data.dataset`;

#2. Write a CASE statement to flag patients as 'High Concern' if they are Extended Stay, and Emergency.

SELECT

Name\_of\_patient

FROM

`nodal-descent-466716-a7.My\_data.dataset`

where Extend\_Stay= 'EXTEND' and EMERGENCY\_OR\_SCHEDULED = 'Emergency';

#3. Get top 3 hospitals with the shortest average stay duration using RANK or DENSE\_RANK.

SELECT

Hospital, RANK() OVER (ORDER BY AVG(Stay\_Days) ASC) AS duration\_rank

FROM

`nodal-descent-466716-a7.My\_data.dataset`

group by hospital

limit 3;

#4.Build a query to pivot Risk Level distribution across hospitals.

SELECT

Hospital,

COUNT(CASE WHEN Risk\_Level = 'Low' THEN 1 END) AS Low\_Risk,

COUNT(CASE WHEN Risk\_Level = 'Medium' THEN 1 END) AS Medium\_Risk,

COUNT(CASE WHEN Risk\_Level = 'High' THEN 1 END) AS High\_Risk

FROM

`nodal-descent-466716-a7.My\_data.dataset`

GROUP BY

Hospital

ORDER BY

Hospital;