

-- REQUIREMENT 1 :CLAIMS FRAUD: HIGH OVERBILLING DETECTION

```
select c.claim_id,p.provider_name,c.claim_amount,m.avg_national_cost,
CASE
WHEN (c.claim_amount/m.avg_national_cost)>=1.5 THEN "HIGH_OVERBILLING"
else "NORMAL_BILLING"
end as Fraud_Flag
```

```
From claims_data c
JOIN provider_master p
ON c.provider_id = p.provider_id
JOIN procedure_code_master m
ON c.procedure_code=m.procedure_code
```

-- REQUIREMENT 2 : PROVIDER BENCHMARK: Rank Billing within each Provider Type (Top 2)

```
with cte2 as (with cte1 as (select p.provider_name,p.provider_type,sum(c.claim_amount) as
total_billed
```

```
from claims_data c
JOIN provider_master p
ON c.provider_id = p.provider_id
group by p.provider_name,p.provider_type)
select * ,
dense_rank()over(partition by provider_type order by total_billed DESC) as Specialty_Rank
from cte1)
select * from cte2
where Specialty_Rank<=2
ORDER BY
```

provider_type, Specialty_Rank;

-- REQUIREMENT 3 : AUDIT FLAG: Provider Count > Threshold for High-Risk Claims

```
with cte1 as (SELECT p.provider_name, m.is_high_risk FROM claims_db.claims_data c
join procedure_code_master m
on c.procedure_code = m.procedure_code
join provider_master p
on c.provider_id = p.provider_id
where m.is_high_risk = "TRUE")
select provider_name, count(is_high_risk) as High_Risk_claim_count from cte1
group by provider_name
having High_Risk_claim_count > 2
order by High_Risk_claim_count DESC
```

-- REQUIREMENT 4 : Cumulative Billing Trend

SELECT

```
p.provider_name,
c.service_date,
c.claim_amount,
-- Window Function for Running Sum (Cumulative Total)
SUM(c.claim_amount) OVER (
ORDER BY c.service_date ASC
) AS cumulative_billed_amount
```

FROM

claims_data c

JOIN

provider_master p

ON p.provider_id = c.provider_id

WHERE

c.provider_id = 4003

ORDER BY

c.service_date ASC;

-- COMPLIANCE CHECK: Flag Claims with Lag > 90 Days

SELECT

c.claim_id,

p.provider_name,

c.service_date,

-- Use the assumed date '2025-12-09' for consistent results

DATEDIFF('2025-12-09', c.service_date) AS Days_Since_Service

FROM

claims_data c

JOIN

provider_master p ON c.provider_id = p.provider_id

-- Filter for non-compliant claims

WHERE

DATEDIFF('2025-12-09', c.service_date) > 90;