2b. Technical support ticket analytics and dashboard Part 2

Status	In progress

Here are few analysis , I have done for SQL Phase 2. Then from here I will be going ahead and building Power BI report and dashboard.

Problem - Reduce time to resolution and improve customer satisfaction by identifying high impact issue categories, agent performance gaps and ticket routing inefficiencies.

```
-- staging table
select * from stg_support_tickets

--
-- How many tickets we received through each channel ?
select
    channel_name,
    COUNT(*) as total_tickets,
    round(COUNT(*) * 100 / SUM(COUNT(*)) over(),2)as percent_tickets
from stg_support_tickets
group by channel_name

/* We see that 79% of tickets are raised through inbound channel, 17% throug
h outcall and 3% through emails. */

-- What are top or top 5 categories/sub categories in each channel that we re
ceive tickets ?
select
    channel_name,
    category,
    sub_category,
```

```
    total_tickets,
    top_category_rank
from
(
select
    channel_name,
    category,
    sub_category,
    COUNT(*) as total_tickets,
    RANK() over(partition by channel_name order by count(*) desc) as top_cate
gory_rank
    from stg_support_tickets
    group by channel_name, category, sub_category
) t
where top_category_rank <=5

/*   */

-- In which channel do customers generally leave a remark ?

select
    channel_name,
    count(*) as total_tickets,
    count(customer_remarks) as total_given_remarks,
    round(count(customer_remarks) * 100 / count(*),2) as percent_remarks
from stg_support_tickets
group by channel_name

/* Customers have almost equally left remarks in all channels with 33% in eac
h channel */

-- Which channel has good average resolution time ?

select
    channel_name,
    CAST(AVG(datediff(MINUTE,issue_reported_date_time, issue_responded_d
```

```
ate_time) / 60.0) as decimal(5,2)) as avg_resolution_hours
from stg_support_tickets
group by channel_name
order by avg_resolution_hours

/*
As we can see that outcall has lowest resolution time whereas email has highe
st. If 2.88 means , 2 hours and multiply 0.88 with 60.
0.88*60 ~ 52 so that becomes 2 hours 52 mins
*/

-- which channel gets better csat score on an average ?

select
   channel_name,
   avg(csat_score) as avg_csat_score
from stg_support_tickets
group by channel_name

--*******--*******--*******--*******--*******--*******--*******--********--*
******--*******--*******--*******--*******

-- How many tickets were created in each category ?
select
   category,
   count(*) as total_ticket
from stg_support_tickets
group by category
order by total_ticket desc

-- What are top 5 ticket types in sub-categories that we get for each product
?  (exclude missing/unknown products) ?

select
   product_category,
   sub_category,
```

```
    total_tickets,
    sub_category_rank
from (
    select
        product_category,
        sub_category,
        count(*) as total_tickets,
        DENSE_RANK() over (partition by product_category order by count(*) des
c) as sub_category_rank
    from stg_support_tickets
    where product_category != 'Product Unknown'
    group by product_category, sub_category
) as t
where sub_category_rank <=5

-- What is the average handling time for differennt issues in sub-category ?

select
    category,
    sub_category,
    round(avg(connected_handling_duration_time),2) as avg_handling_seconds,
    round(avg(connected_handling_duration_time)/60,2) as avg_handling_minut
es
from stg_support_tickets
group by category,sub_category
order by avg_handling_minutes desc
--*******--*******--*******--*******--*******--*******--*******--********--*
******--*******--*******--*******--*******

-- what are different cities we get customers reach out to us ?
-- count of total different cities
select COUNT(distinct customer_city) as total_cities from stg_support_tickets
-- ~ 1783
-- what are different cities we get customers reach out to us ?
select distinct customer_city from stg_support_tickets order by customer_city
```

```
-- Which cities customers reached out with how many tickets ?
-- top 10 cities with high ticket recorded ?
select top 10
   customer_city,
   count(*) as total_tickets
from stg_support_tickets
where customer_city != 'Location Unknown'
group by customer_city
order by total_tickets desc

-- Top cities / top 10 cities with high purchase value?
select top 10
   customer_city,
   sum(item_price) as total_purchase_value
from stg_support_tickets
where customer_city != 'Location Unknown'
group by customer_city
order by total_purchase_value desc
--*******--*******--*******--*******--*******--*******--*******--********--*
******--*******--*******--*******--*******
-- How many differenct products we have ?
select distinct
   product_category
from stg_support_tickets

-- How many tickets are recorded for each product that we have ?

select distinct
   product_category,
   count(*) as total_tickets
from stg_support_tickets
group by product_category
order by total_tickets desc
--*******--*******--*******--*******--*******--*******--*******--********--*
******--*******--*******--*******--*******
-- What is the average resolution time ?
```

```
select
   avg(DATEDIFF(MINUTE, issue_reported_date_time, issue_responded_date_ti
me)) as avg_resolution_time_minutes,
   cast(avg(DATEDIFF(MINUTE, issue_reported_date_time, issue_responded_d
ate_time))/60.0 as decimal(5,2)) as avg_resolution_time_hours
from stg_support_tickets

-- List managers whose team have got low resolution time
select
   manager,
   cast(avg(DATEDIFF(MINUTE, issue_reported_date_time, issue_responded_d
ate_time))/60.0 as decimal(5,2)) as avg_resolution_time_hours
from stg_support_tickets
group by manager
order by avg_resolution_time_hours asc

-- flag tickets resolved and not resolved under 48 hours SLA ?
select *
from
(
select
   *,
   cast(DATEDIFF(minute, issue_reported_date_time, issue_responded_date_ti
me)/60 as decimal(5,2)) as resolution_time,
   case
      when cast(DATEDIFF(minute, issue_reported_date_time, issue_responded
_date_time)/60 as decimal(5,2)) <= 48 THEN 'SLA GOOD' ELSE 'SLA BAD'
   END AS sla_flag
from stg_support_tickets
) as t
where sla_flag = 'SLA BAD'

-- Number of tickets and Percent of tickets that met sla and that did not meet
sla
SELECT
   SUM(CASE WHEN sla_flag = 'SLA GOOD' THEN 1 ELSE 0 END) AS tickets_sl
```

```
a_good,
   SUM(CASE WHEN sla_flag = 'SLA BAD' THEN 1 ELSE 0 END) AS tickets_sla
_bad,
   cast(SUM(CASE WHEN sla_flag = 'SLA GOOD' THEN 1 ELSE 0 END) * 100.0
/ COUNT(*) as decimal(5,2)) AS sla_good_percent,
   cast(SUM(CASE WHEN sla_flag = 'SLA BAD' THEN 1 ELSE 0 END) * 100.0 /
COUNT(*) as decimal(5,2)) AS sla_bad_percent
FROM (
   SELECT
      CASE
         WHEN DATEDIFF(MINUTE, issue_reported_date_time, issue_responded
_date_time) / 60.0 <= 48 THEN 'SLA GOOD'
         ELSE 'SLA BAD'
      END AS sla_flag
   FROM stg_support_tickets
) AS t

--*******--*******--*******--*******--*******--*******--*******--********--*
******--*******--*******--*******--*******
-- Check if when item price is high then resolution time taken is low ?

select top 10
   category,
   sub_category,
   product_category,
   issue_reported_date_time,
   issue_reported_date_time,
   item_price,
   cast(DATEDIFF(MINUTE, issue_reported_date_time, issue_responded_date_
time) / 60.0 as decimal(5,2)) as resolution_time_hours
from stg_support_tickets
where item_price is not null
order by item_price desc


select distinct
```

```
item_price
from stg_support_tickets
order by item_price

-- Check if when item price is high then resolution time taken is low ?
select
   case
      when item_price >= 25000 then 'High Price'
      when item_price between 5000 and 24999 then 'Medium Price'
      else 'Low Price'
   end as price_band,
   count(*) as total_tickets,
   cast(avg(DATEDIFF(minute, issue_reported_date_time, issue_responded_da
te_time)/60)as decimal(5,2)) as avg_resolution_hours,
   avg(csat_score) as csat_score
from stg_support_tickets
where item_price is not null
group by
   case
      when item_price >= 25000 then 'High Price'
      when item_price between 5000 and 24999 then 'Medium Price'
      else 'Low Price'
   end
order by avg_resolution_hours
--*******--*******--*******--*******--*******--*******--*******--********--*
******--*******--*******--*******--*******
-- count of agents, supervisors and managers
select
count(distinct agent_name) as total_agents,
count(distinct supervisor) as total_sups,
count(distinct manager) as total_managers
from stg_support_tickets

-- Number of supervisors and agents under one manager
select
   manager,
```

```
    count(distinct supervisor) as total_sups,
    count(distinct agent_name) as total_agents
from stg_support_tickets
group by manager
order by total_agents asc

select
    manager,
    supervisor,
    count(distinct agent_name) as total_agents
from stg_support_tickets
group by manager, supervisor
order by manager asc, total_agents desc

-- How is resolution time and call handling time with respect to tenurity, are m
ore tenure taking less time ?
-- How is resolution time and CSAT with respect to tenurity, are more tenure t
aking less time ?

select
    tenure,
    cast(avg(DATEDIFF(minute, issue_reported_date_time, issue_responded_da
te_time)/60)as decimal(5,2)) as avg_resolution_hours,
    cast(avg(connected_handling_duration_time)/60 as decimal(5,2)) as avg_ha
ndling_time_minutes,
    avg(csat_score) as avg_csat
from stg_support_tickets
group by tenure
order by tenure desc

-- How is shift vs CSAT score ?

select
    agent_shift,
    cast(avg(DATEDIFF(minute, issue_reported_date_time, issue_responded_da
te_time)/60)as decimal(5,2)) as avg_resolution_hours,
```

```
    cast(avg(connected_handling_duration_time)/60 as decimal(5,2)) as avg_ha
ndling_time_minutes,
    avg(csat_score) as avg_csat
from stg_support_tickets
group by agent_shift
```