AXIS DATA ANALYTICS

MONTHLY COMMISSION REPORT PROJECT FOR WILD WEST AB

GROUP 9

DATE : 14/11/2024

About Axis Data Analytics

Axis Data Analytics is a leading provider of data-driven insights and analytics solutions, specializing in transforming raw data into actionable business intelligence. We offer comprehensive services that cater to a wide range of industries, with a focus on insurance data analysis, business optimization, and strategic decision-making. Our team of experts helps clients unlock the power of data to drive innovation, improve efficiency, and achieve growth.

At Axis, we use advanced analytical techniques and cutting-edge technologies to deliver accurate, reliable, and insightful reports tailored to meet the unique needs of each client.

Meet the Axis Data Analytics Project Team

- **Praveen Mohan** - Lead Analyst
- **William Hoang** - Data Analyst
- **Wei Wang** - QA Analyst
- **Joseph Richardson** - Reporting Specialist

Optimized Data Model for Efficient Reporting and Analysis

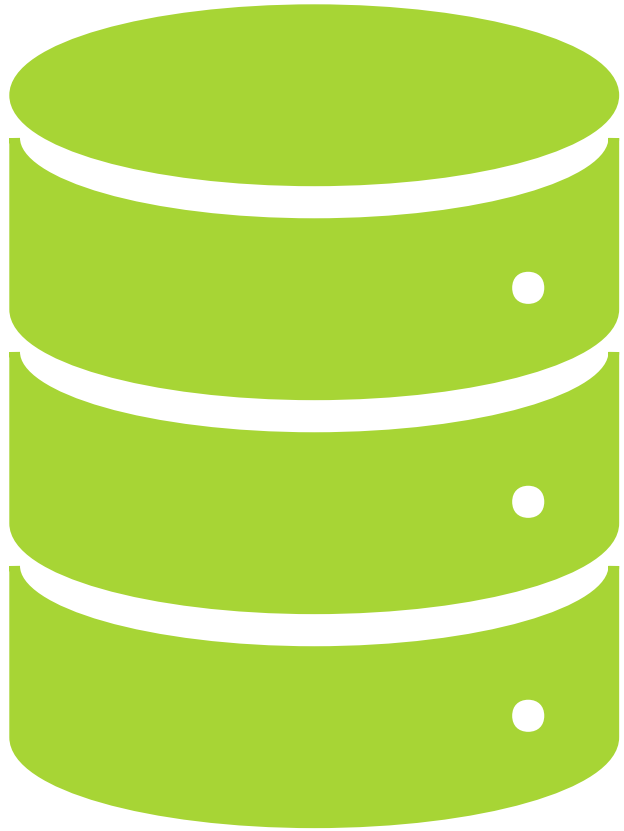
- **Efficient Relationships** : The table summarizing all relationships, including 1-to-Many and Many-to-Many connections, along with supporting tables and their descriptions

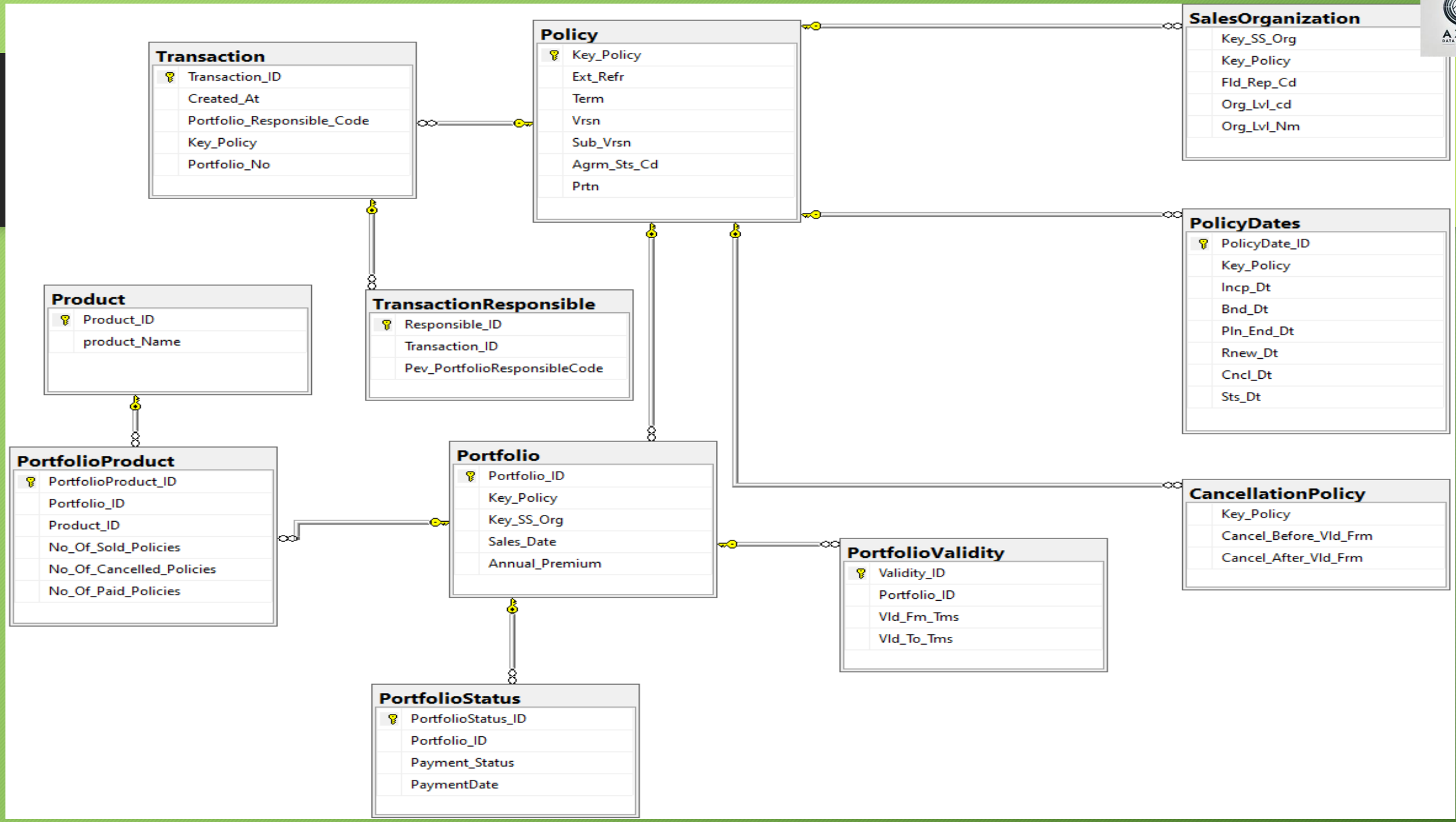
- This data model is structured in Third Normal Form (3NF), ensuring that each table serves a distinct purpose and minimizes data redundancy.

Parent Table	Child Table	Relationship Type	Junction Table	Description
Portfolio	Policy	1-to-Many	-	A single portfolio can include multiple policies, but each policy is linked to only one portfolio.
Policy	Policy Transactions	1-to-Many	-	Each policy can have multiple transactions (e.g., payments, claims), but each transaction is linked to one policy only.
Policy	PolicyDates	1-to-Many	-	Each policy is associated with multiple key dates (e.g., inception, renewal), but each date entry pertains to one policy.
Portfolio	Portfolio Status	1-to-Many	-	A portfolio canundergo multiple status changes (e.g., active, inactive) over time, but each status entry is for one portfolio.
Portfolio	Product	Many-to-Many	PortfolioProduct	A portfolio can include multiple products, and each product can appear in multiple portfolios, managed via PortfolioProduct.
Policy	Agent	Many-to-Many	PolicyAgent	Policies can have multiple agents, and agents can manage multiple policies, handled through the PolicyAgent junction table.
Supporting Table	Transaction Types	-	-	Defines types of transactions (e.g., payment, refund) for standardized use in Policy Transactions.
Supporting Table	Policy Statuses	-	-	Lists all possible statuses (e.g., active, inactive) for policies, enabling consistent status tracking across policies.
Supporting Table	PolicyDates	-	-	Records important dates related to policies (e.g., inception, renewal) separate from the core policy data for clarity.

3 NF DATA MODEL

- This data model is both robust and efficiently designed, with proper normalization ensuring consistency across the dataset. The integration of unique identifiers (IDs) enhances accuracy and optimizes the model for efficient querying and reporting. By reducing redundancy, we've significantly improved data integrity. Overall, this structure is well-suited for reliable, streamlined reporting and analytics, providing a solid foundation for decision-making and future insights.




Transaction
Transaction_ID
Created_At
Portfolio_Responsible_Code
Key_Policy
Portfolio_No
Policy
Key_Policy
Ext_Refr
Term
Vrsn
Sub_Vrsn
Agrm_Sts_Cd
Prtn
SalesOrganization
Key_SS_Org
Key_Policy
Fld_Rep_Cd
Org_Lvl_cd
Org_Lvl_Nm
PolicyDates
PolicyDate_ID
Key_Policy
Incp_Dt
Bnd_Dt
Pln_End_Dt
Rnew_Dt
Cncl_Dt
Sts_Dt
Product
Product_ID
product_Name
TransactionResponsible
Responsible_ID
Transaction_ID
Pev_PortfolioResponsibleCode
PortfolioProduct
PortfolioProduct_ID
Portfolio_ID
Product_ID
No_Of_Sold_Policies
No_Of_Cancelled_Policies
No_Of_Paid_Policies
Portfolio
Portfolio_ID
Key_Policy
Key_SS_Org
Sales_Date
Annual_Premium
CancellationPolicy
Key_Policy
Cancel_Before_Vld_Frm
Cancel_After_Vld_Frm
PortfolioValidity
Validity_ID
Portfolio_ID
Vld_Fm_Tms
Vld_To_Tms
PortfolioStatus
PortfolioStatus_ID
Portfolio_ID
Payment_Status
PaymentDate

COMMISSION ELIGIBILITY CRITERIA

1. The policy has been marked with the portfolio code as ‘WILDWEST-2’ or
‘WILDWEST-3’
2. The policy has been marked with the portfolio code for the entire duration of its
existence.
3. The premium for the policy is paid.
4. The following products are within the scope of the agreement. No other product
sold even with the portfolio code, will be valid for a commission:
a. ‘Product 1’ through ‘Product 8’
b. ‘Product 13’ through ‘Product 31’
5. Moreover, the policy sale should fulfil the following criteria:
a. Outbound sales: portfolio code = ‘WILDWEST-3’
b. Internet Sales: portfolio code = ‘WILDWEST-3’
c. Inbound Sales: portfolio Code = ‘WILDWEST-2’
No other sales channel would be eligible for commissions.

The commission is paid as per the following rules:
1. For the first 1500 policies, the commission is 12% of the annual premium.
2. From the 1501-th policy, the commission is 14% of the annual premium.

STRATEGIES TO BOOST COMMISSION-ELIGIBLE POLICIES FOR AGENTS

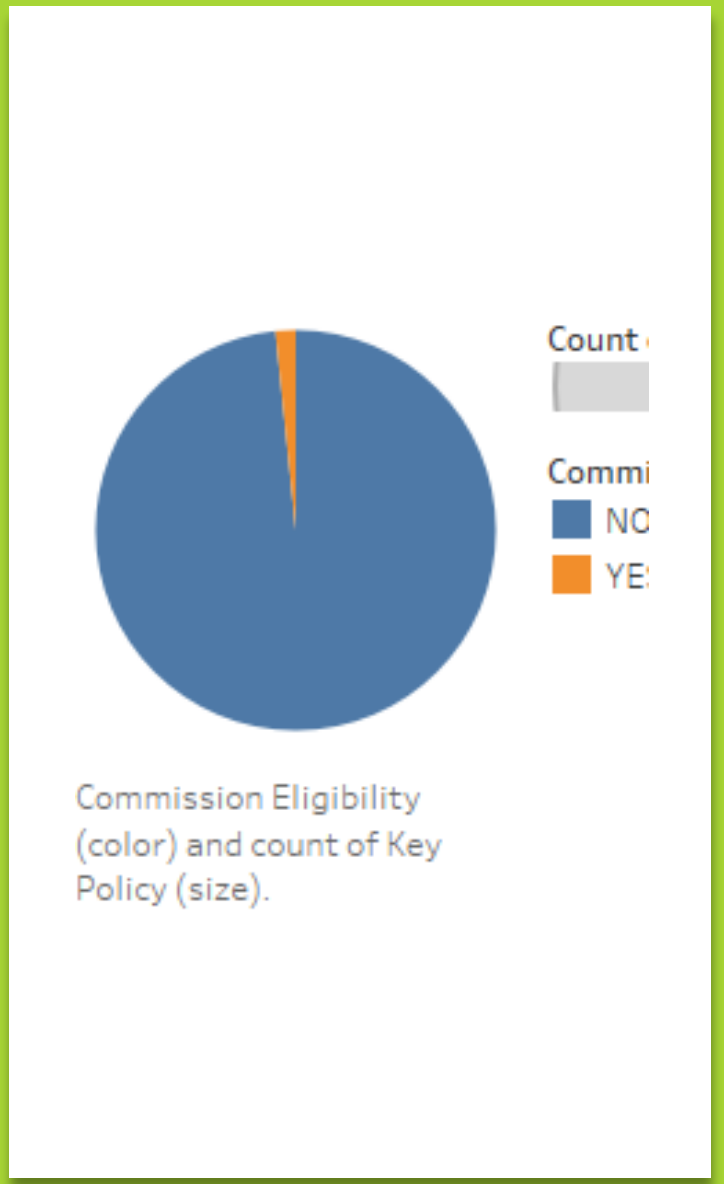
1 → 2 → 3 → 4

Incentivize Desired Policy Types : Offer additional incentives or bonuses to agents for selling policies with high commission eligibility potential.

Adjust Commission Criteria : Relax commission eligibility criteria to make it more attainable for policies that support business goals like retention and customer satisfaction.

Training and Performance Support : Provide targeted training to agents on selling commission-eligible policies, especially those with complex eligibility criteria.

Promote Long-Term Policies : Encourage customers to choose long-term policies by highlighting benefits like stability and long-term support.



The strategies are aimed at increasing the number of commission-eligible policies that agents can sell, which would directly impact their earnings and incentivize them to focus on selling more qualifying policies.

Assumptions for Wild West Policy Analysis

Data Scope

Limited the dataset to only the last 8 months.

This timeframe was chosen for analysis and reporting, possibly to focus on recent data.

Date Assumption

Selected two date fields to work with: ”Vld_Fm_tms - From time" (assumed as the start date) and "VLD - To time" (assumed as the end date).

Policy Duration Calculation

Added a new column labeled "Policy Duration".

The policy duration is calculated as the difference between "VLD - From time" and "VLD - To time".

This new column provides the length of time for each policy.

Handling Null Cancellation Dates:

For policies with no cancellation date (i.e., cancellation date is NULL), assumed the "Policy Status Date – 1 " as the cancellation date.

This assumption allows for consistent handling of policies with missing cancellation dates.

Used the "Cancel Status" code from the "Agrm - Sts Cd" field to identify canceled policies.

Policy End Date as Accurate Representation of Termination

The `Vld_To_tms` field represents the definitive end date for each policy, accurately reflecting cancellations or expirations without requiring secondary checks.

Policy Type Classification

Added a new column titled "Policy Type" to classify each policy as either Premium, Standard, or Long-term.

Premium: If the Annual Premium is greater than or equal to 1500.

Long-term: If the policy duration is more than 365 days.

Standard: If neither Premium nor Long-term criteria are met.

Assumptions for Wild West Policy Analysis

Data Consistency in Versioning

It's assumed that `Vrsn` and `Sub_Vrsn` fields reliably indicate the latest policy version, meaning only the most recent policy state will be selected for each unique policy.

Agent Performance Rating

Created a new column named "Agent Performance" to rate agent performance as either Good or Bad.

Rating criteria:

Good: If the Annual Premium is greater than or equal to 1000.

Bad: If the Annual Premium is less than 1000.

Season Classification

Added a new column to categorize policies by season based on the policy effective dates.

Seasonal categories:

Winter: December to February.

Spring: March to May.

Summer: June to August.

Fall: September to November.

Exclusion of First Month for Calculations

Assumed that no commission or clawback calculations would be made in the first month of the policy.

This may be due to a waiting period or policy setup time.

Active vs. Canceled Policy Status

Assumed that policies with durations exceeding 365 days are Active.

Policies with durations less than or equal to 365 days are considered in Cancel Status.

This provides a clear separation between long-term and potentially short-term or canceled policies.

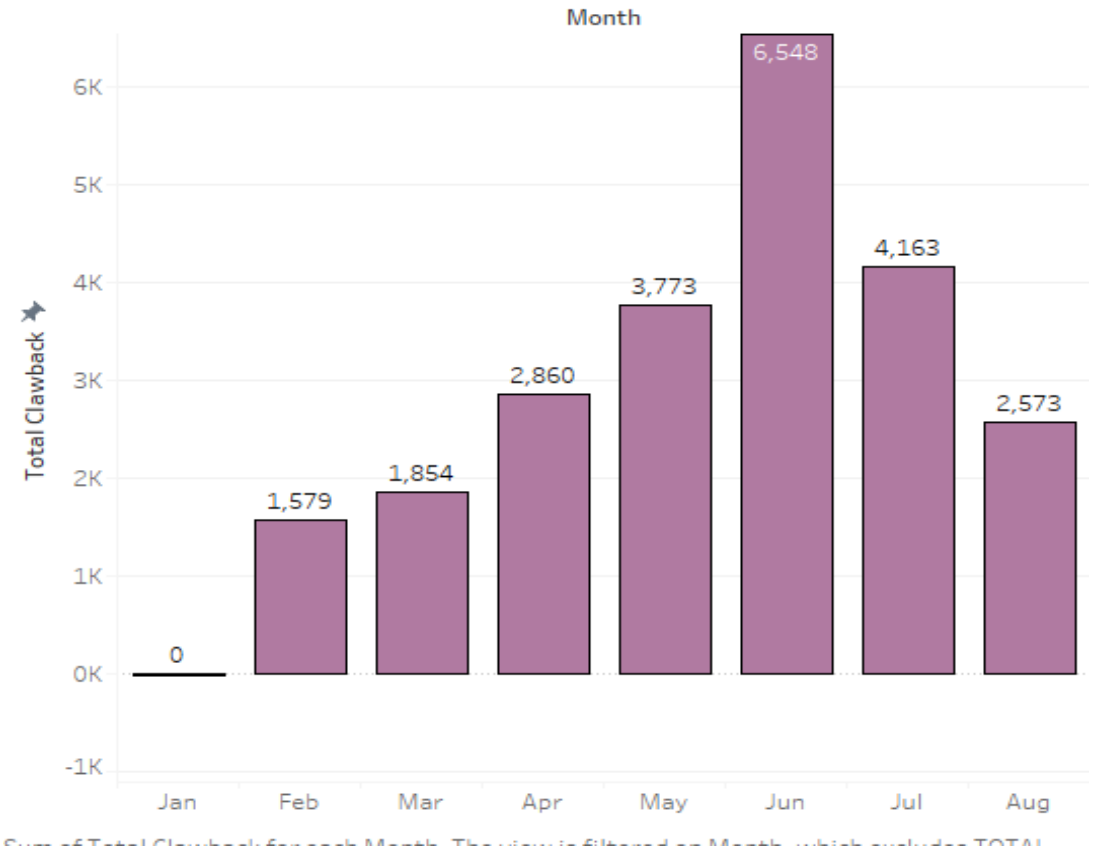
Commission Eligibility Column

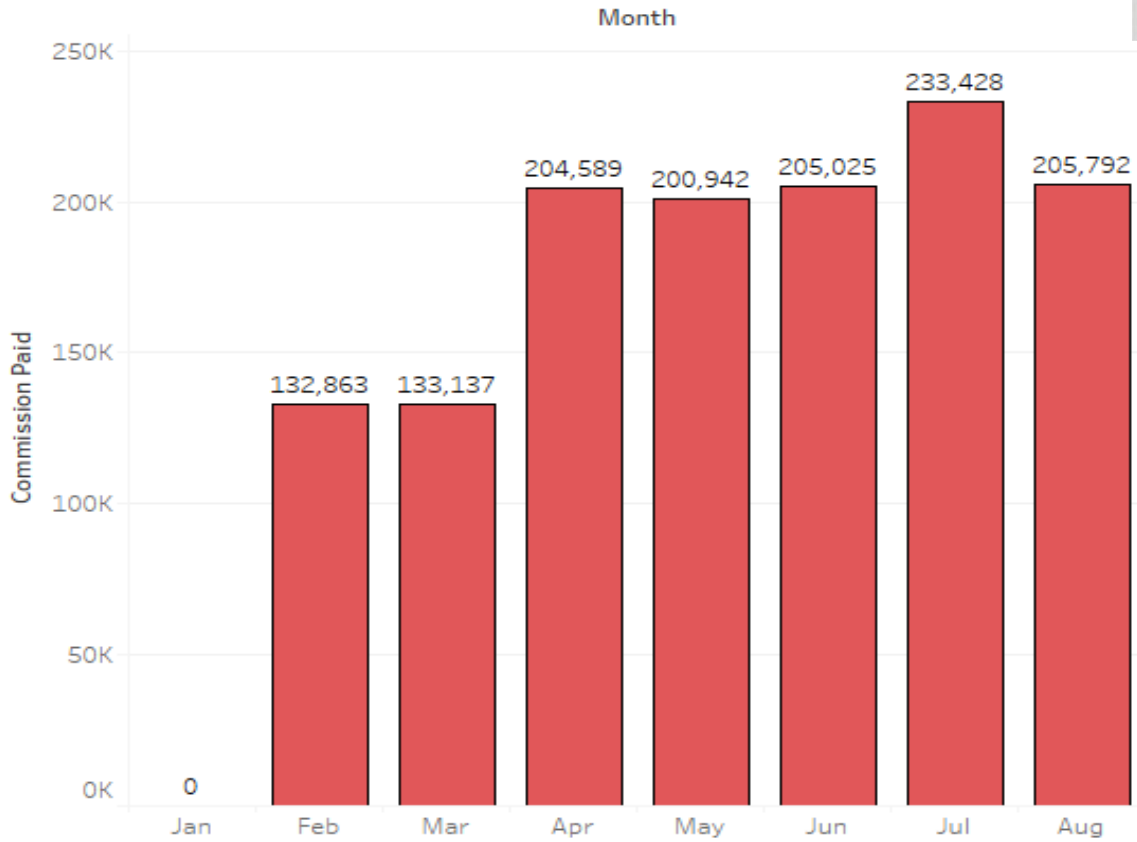
Added a new column called "Commission Eligibility" to enhance clarity around which policies qualify for commission and clawback calculations.

This column provides a quick reference for determining eligibility. Commission and clawback calculations were performed only if the "Commission Eligibility" status is marked as "Yes".

Mont	Total Policie	Active Policie	Cancelled Policie	Premium Revenu	Commission Pai	Total Clawbac
Jan	243	115	128	1133693	0	0
Feb	213	91	122	1107189	132862.68	1578.77
Mar	322	135	187	1109472	133136.64	1854.38
Apr	464	199	265	1704909	204589.08	2860.03
May	479	177	302	1665239	200942.44	3772.55
Jun	399	156	243	1464464	205024.96	6547.64
Jul	451	199	252	1667342	233427.88	4163.07
Aug	391	172	219	1469945	205792.3	2572.77
TOTAL	2962	1244	1718	1,13,22,253	13,15,775.98	23,349.21

AGGREGATED MONTHLY POLICY PERFORMANCE (JAN–AUG 2024)

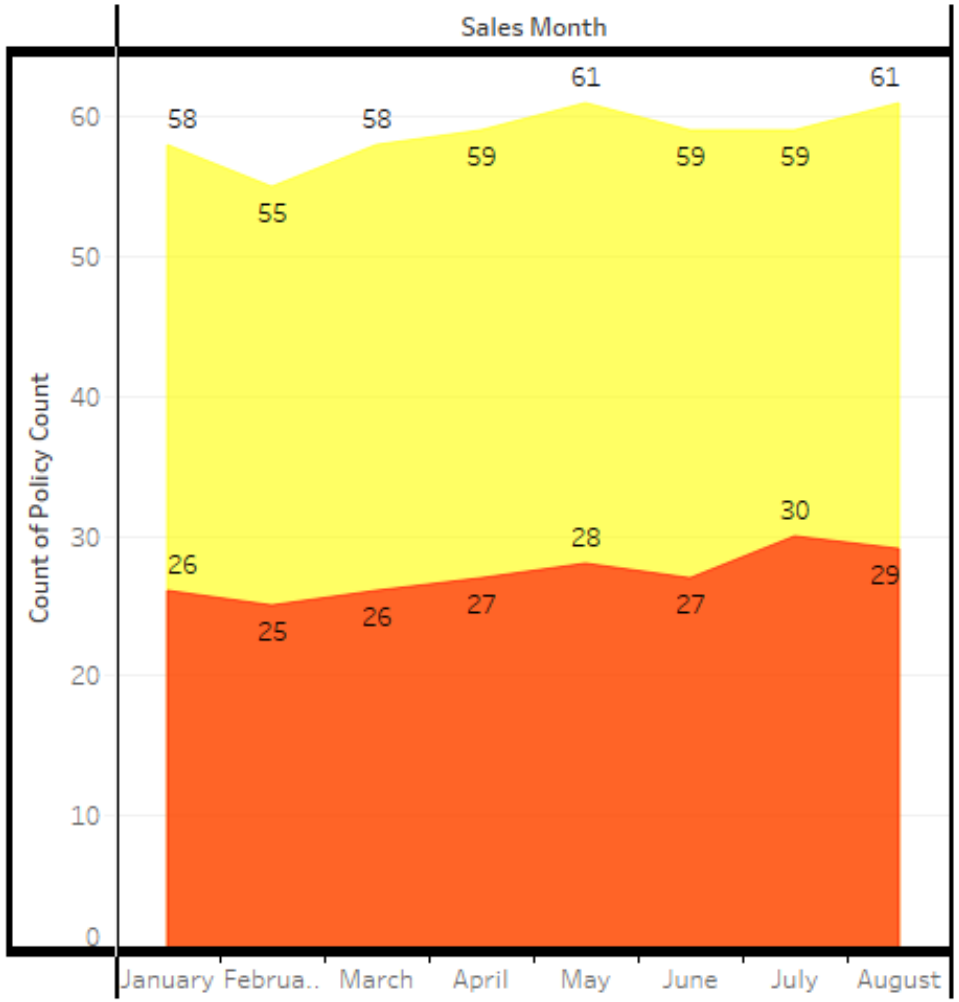




TRENDS IN COMMISSION PAYOUTS VS. CLAWBACKS

ADDITIONAL INSIGHTS

ACTIVE VS CANCEL POLICIES



INSIGHTS INTO POLICY RETENTION ISSUES

Gradual Increase in Cancellations: There is a steady increase in cancellations over the months, peaking in July. This indicates a potential issue with customer retention as time progresses.

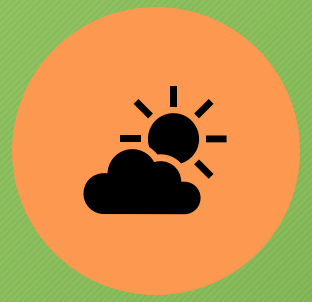
Seasonal Fluctuations in Cancellations: The chart hints at possible seasonal impacts on policy cancellations, which may suggest that some months or periods of the year have a higher risk of customer churn.

Inconsistent Active Policy Growth: While active policies are high, they do not show consistent growth across months. This could indicate missed opportunities in boosting active policies further.

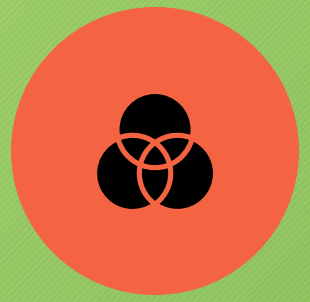
Lack of Clear Strategies to Curb Mid-Year Cancellations: The uptick in cancellations in mid-year (July and August) suggests that retention efforts may not be as effective during this period, leading to preventable losses.

STRATEGIES TO ENHANCE POLICY RETENTION

1. Launch Mid-Year Retention Campaigns : Implement targeted retention campaigns during mid-year months (especially July and August) to incentivize policyholders to stay active, using loyalty rewards, benefit reminders, or special offers for policy extensions.



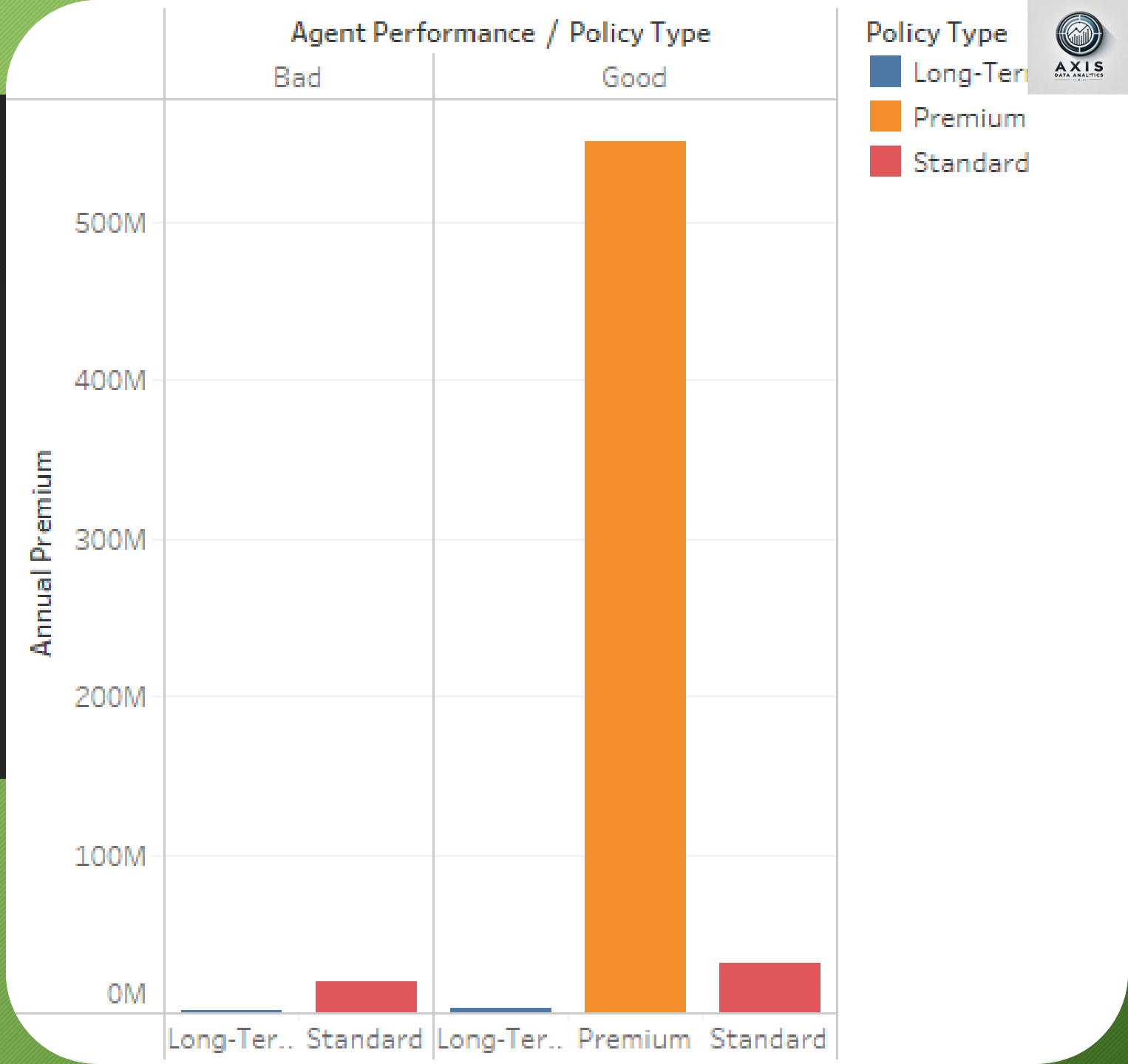
2. Deepen Seasonal Analysis : Examine seasonal patterns to understand why cancellations peak in certain months, and address specific external factors (like vacation timing or financial cycles) with tailored offers or incentives.



3. Introduce Flexible Policy Options : Offer flexibility, such as premium adjustments or grace periods, that make it easier for policyholders to stay active during financially challenging times.

4. Incentivize Early Renewals : Encourage policy renewals ahead of high-cancellation periods by offering discounts or incentives to retain customers before they consider canceling.

AGENT PERFORMANCE SEGMENTATION BY POLICY TYPE



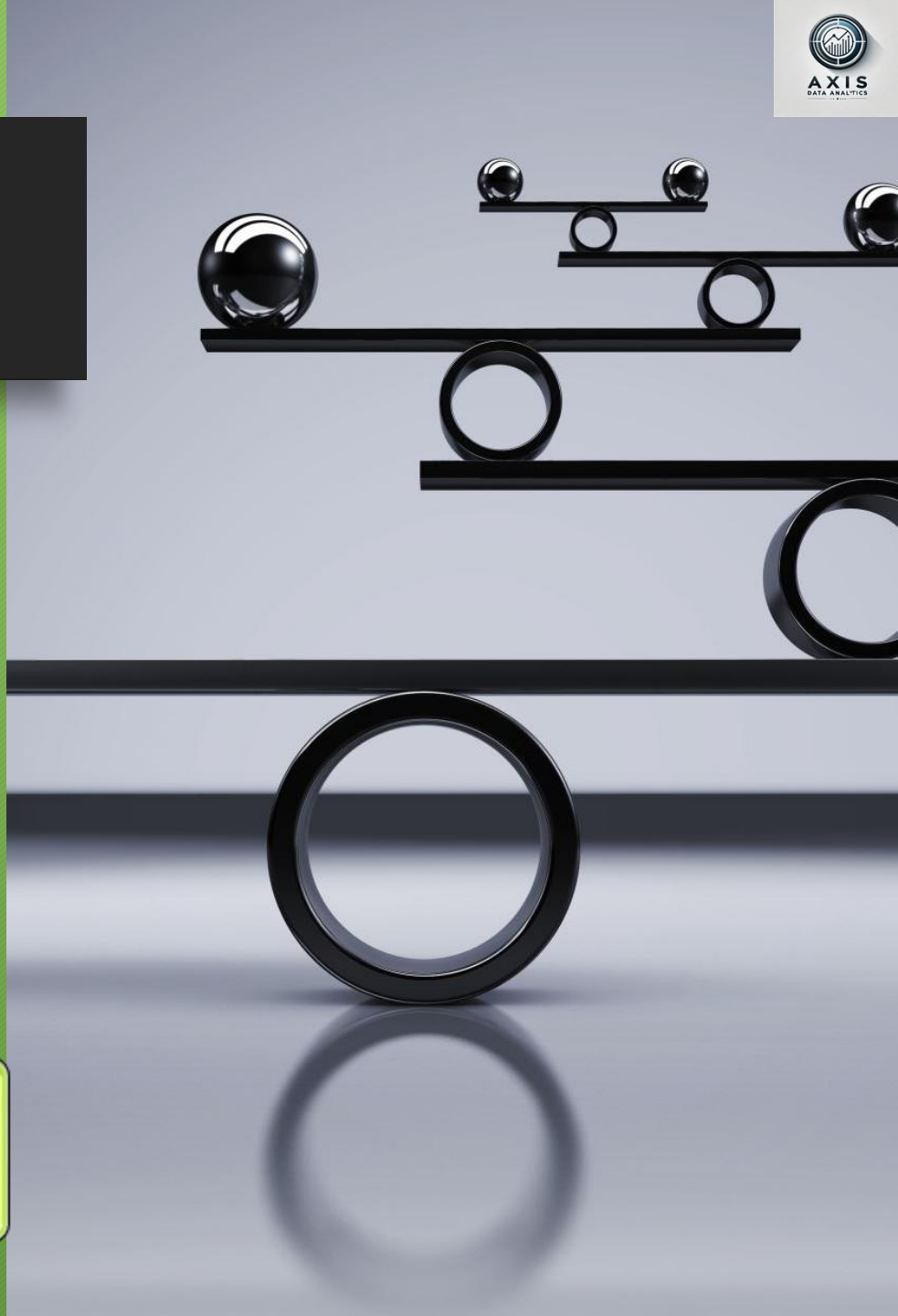
INSIGHTS INTO PERFORMANCE GAPS

Imbalance in Policy Types : Premium policies dominate, limiting diversification across other types.

Underperformance in "Bad" Agents : Low annual premiums for "Bad" agents suggest potential inefficiencies or misalignment in sales strategy.

Limited Long-Term Policies : Minimal representation of Long-Term policies reduces stability in the revenue stream.

Standard Policies Underutilized : Low premiums from Standard policies indicate missed opportunities for broader customer reach.



APPROACHES TO STRENGTHEN PERFORMANCE

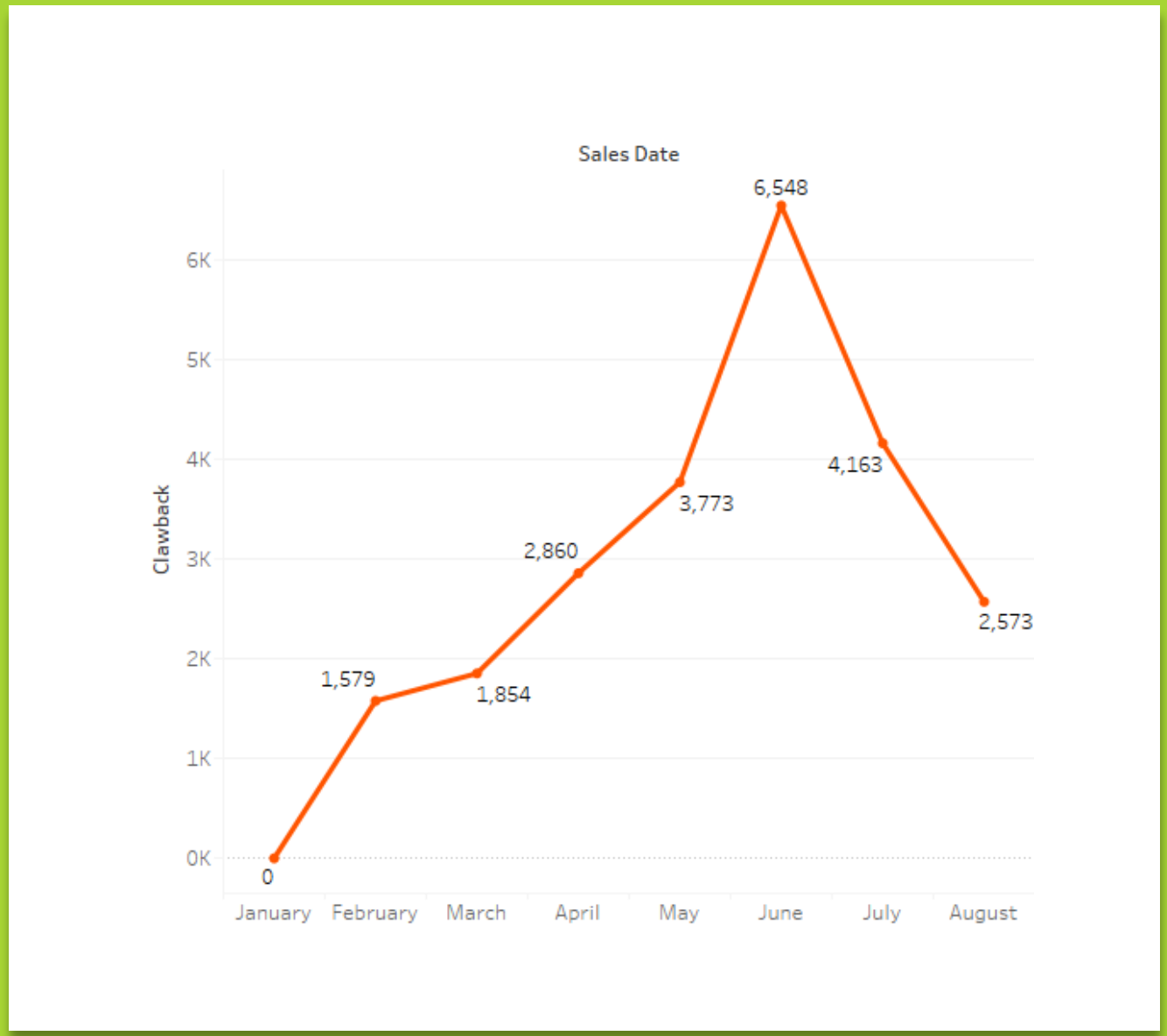
Promote Standard Policies : Create incentives to boost sales of Standard policies for a balanced portfolio.

Targeted Training for "Bad" Agents : Offer performance-improvement programs focused on increasing efficiency and premium generation.

Diversify Product Offerings : Actively promote Long-Term policies to enhance customer retention and consistent revenue.

Refocus Premium Policy Sales Efforts : Optimize marketing strategies for Premium policies while ensuring sustainable sales growth across all types.

MONTHLY CLAWBACK TRENDS IN 8-MONTH SALES CYCLE



Challenges in Managing Clawback Trends

- **Fluctuating Clawback Values**: The significant month-to-month changes suggest unstable clawback trends, which may indicate inconsistencies in policy retention.
- **High Mid-Year Peak**: The spike in June may imply increased cancellations, affecting overall revenue and raising red flags for underlying causes.
- **Declining Clawback Post-June**: The steep decline after June could suggest a seasonal pattern, which might affect long-term financial planning.
- **Limited Data Window**: Analyzing only eight months may not provide a full picture, potentially missing broader trends affecting clawback behavior.

Strategies to Mitigate Clawback Challenges

Implement Customer Retention Programs: Target customer support and engagement during high clawback months to reduce cancellations.

Analyze Causes of Mid-Year Spike: Investigate the reasons for the June peak to address policy or service issues leading to increased clawbacks.

Develop Seasonality Models: Create predictive models that consider seasonality to better manage clawback rates and improve forecasting accuracy.

Expand Data Range: Gather and analyze data beyond eight months to capture a more comprehensive view of trends and improve strategic decision-making.

Concluding Summary and Actionable Insights

- In this analysis of WildWest AB policy data, we successfully calculated commission and clawback metrics in alignment with project requirements, providing a clear view of financial performance across policies. Additionally, our insights revealed key patterns, such as the mid-year spike in clawback values, highlighting areas for improvement in policy retention strategies. This data-driven approach not only fulfills the primary objectives but also uncovers actionable trends, enabling Axis to implement more targeted retention strategies and optimize financial outcomes. By continuously refining our analysis, we can support sustainable growth and improved customer retention in the WildWest AB portfolio.

THANK YOU

THIS IS THE MAIN CODE

```
USE [Wild West];

-- Step 1: Create a new table 'Month_Data_8' and insert January data into it
SELECT *
INTO dbo.Month_Data_8  -- Create the table 'Month_Data_8' and insert data for January 2024
FROM Portfolio_clean
WHERE MONTH(Sales_Date) = 1;

-- Step 2: Insert data for February to August into the existing 'Month_Data_8' table
INSERT INTO dbo.Month_Data_8
SELECT *
FROM Portfolio_clean
WHERE MONTH(Sales_Date) IN (2, 3, 4, 5, 6, 7, 8);

-- Step 3: Ensure that the target table has the correct structure
-- Add 'Commission' column if it doesn't exist
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.COLUMNS
        WHERE TABLE_NAME = 'Month_Data_8' AND COLUMN_NAME = 'Commission')
BEGIN
  ALTER TABLE dbo.Month_Data_8
  ADD Commission DECIMAL(18, 2);
END;

-- Add 'Commission_Eligibility' column if it doesn't exist
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.COLUMNS
        WHERE TABLE_NAME = 'Month_Data_8' AND COLUMN_NAME =
'Commission_Eligibility')
BEGIN
```

```
  ALTER TABLE dbo.Month_Data_8
  ADD Commission_Eligibility VARCHAR(3);
END;

-- Add 'Agm_Sts_Cd' column if it doesn't exist
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.COLUMNS
        WHERE TABLE_NAME = 'Month_Data_8' AND COLUMN_NAME =
'Agm_Sts_Cd')
BEGIN
  ALTER TABLE dbo.Month_Data_8
  ADD Agm_Sts_Cd VARCHAR(50);
END;

-- Add 'Org_Lvl_Nm' column if it doesn't exist
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.COLUMNS
        WHERE TABLE_NAME = 'Month_Data_8' AND COLUMN_NAME =
'Org_Lvl_Nm')
BEGIN
  ALTER TABLE dbo.Month_Data_8
  ADD Org_Lvl_Nm VARCHAR(50);
END;

-- Add 'Fld_Rep_Cd' column if it doesn't exist
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.COLUMNS
        WHERE TABLE_NAME = 'Month_Data_8' AND COLUMN_NAME = 'Fld_Rep_Cd')
BEGIN
  ALTER TABLE dbo.Month_Data_8
  ADD Fld_Rep_Cd VARCHAR(50);
END;

-- Add 'pev_portfolioResponsibleCode' column if it doesn't exist
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.COLUMNS
```

```
        WHERE TABLE_NAME = 'Month_Data_8' AND COLUMN_NAME =
'pev_portfolioResponsibleCode')
BEGIN
  ALTER TABLE dbo.Month_Data_8
  ADD pev_portfolioResponsibleCode VARCHAR(50);
END;

-- Add 'pev_id' column if it doesn't exist
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.COLUMNS
        WHERE TABLE_NAME = 'Month_Data_8' AND COLUMN_NAME = 'pev_id')
BEGIN
  ALTER TABLE dbo.Month_Data_8
  ADD pev_id VARCHAR(50);
END;

-- Add 'policy_type' column if it doesn't exist
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.COLUMNS
        WHERE TABLE_NAME = 'Month_Data_8' AND COLUMN_NAME = 'policy_type')
BEGIN
  ALTER TABLE dbo.Month_Data_8
  ADD policy_type VARCHAR(50);
END;

-- Add 'agent_performance' column if it doesn't exist
IF NOT EXISTS (SELECT * FROM INFORMATION_SCHEMA.COLUMNS
        WHERE TABLE_NAME = 'Month_Data_8' AND COLUMN_NAME =
'agent_performance')
BEGIN
  ALTER TABLE dbo.Month_Data_8
  ADD agent_performance VARCHAR(50);
END;
```

```
-- Step 4: Add 'Policy_Status' column if it doesn't already exist
IF COL_LENGTH('dbo.Month_Data_8', 'Policy_Status') IS NULL
BEGIN
  ALTER TABLE dbo.Month_Data_8
  ADD Policy_Status VARCHAR(10);
END;

-- Step 5: Add necessary columns to 'Month_Data_8' if they don't already exist
-- Add 'Season' column to store the season (Q1, Q2, Q3, Q4)
IF COL_LENGTH('dbo.Month_Data_8', 'Season') IS NULL
BEGIN
  ALTER TABLE dbo.Month_Data_8
  ADD Season VARCHAR(5);
END;

-- Add 'Seasonal_Performance' column to store performance indicator
IF COL_LENGTH('dbo.Month_Data_8', 'Seasonal_Performance') IS NULL
BEGIN
  ALTER TABLE dbo.Month_Data_8
  ADD Seasonal_Performance VARCHAR(20);
END;

-- Ensure 'Clawback_Category' column exists in 'Month_Data_8'
IF COL_LENGTH('dbo.Month_Data_8', 'Clawback_Category') IS NULL
BEGIN
  ALTER TABLE dbo.Month_Data_8
  ADD Clawback_Category VARCHAR(20); -- To store clawback timing categories like 'Early', 'Mid-Term', 'Late'
END;

-- Step 6: Update 'Policy_Status' based on 'Policy_Duration' in 'Month_Data_8'
UPDATE dbo.Month_Data_8
```

```
SET Policy_Status = CASE
  WHEN Policy_Duration >= 365 THEN 'Active'
  ELSE 'Cancel'
END;

-- Step 7: Create a temporary table to gather policy and transaction details
WITH PolicyTransactions AS (
  SELECT
    p.Key_Policy,
    p.bnd_dt,
    p.pln_end_dt,
    p.Vld_Fm_Tms,
    p.Vld_To_Tms,
    p.Agrm_Sts_Cd,
    pt.pev_createdat,
    pt.pev_portfolioresponsiblecode,
    pt.pev_id,
    p.Vrsn,
    p.Sub_Vrsn,
    DATEDIFF(day, p.Vld_Fm_Tms , p.Vld_To_Tms) AS Policy_Duration,
    DATEADD(day, -1, p.Sts_Dt) AS Cncl_Dt
  FROM dbo.Policy_clean p
  JOIN dbo.Policy_Transactions_clean pt ON p.Ext_Refr = pt.po_No
  WHERE
    p.Agrm_Sts_Cd IN ('Active', 'Cancel')
    AND p.Vld_Fm_Tms BETWEEN '2024-01-01' AND '2024-08-31'
    AND p.Vld_To_Tms IS NOT NULL
    AND p.Vrsn = (SELECT MAX(Vrsn) FROM dbo.Policy_clean WHERE Key_Policy =
p.Key_Policy)
    AND p.Sub_Vrsn = (SELECT MAX(Sub_Vrsn) FROM dbo.Policy_clean WHERE
Key_Policy = p.Key_Policy AND Vrsn = p.Vrsn)
    AND pt.pev_createdat BETWEEN p.Vld_Fm_Tms AND p.Vld_To_Tms
```

```
),

-- Step 8: Filter consistent portfolio codes for 'WILDWEST-2' and 'WILDWEST-3'
ConsistentPortfolioCodes AS (
  SELECT Key_Policy, MIN(pev_portfolioresponsiblecode) AS PortfolioCode
  FROM PolicyTransactions
  GROUP BY Key_Policy
  HAVING COUNT(DISTINCT pev_portfolioresponsiblecode) = 1
    AND MIN(pev_portfolioresponsiblecode) IN ('WILDWEST-2', 'WILDWEST-3')
),

-- Step 9: Prepare 'SalesPortfolio' with relevant policy data
SalesPortfolio AS (
  SELECT
    po.Key_Policy,
    po.Payment_Status,
    po.Product_Name,
    po.Sales_Date,
    po.Vld_Fm_Tms,
    po.Vld_To_Tms,
    DATEDIFF(day, po.Vld_Fm_Tms, po.Vld_To_Tms) AS Policy_Duration,
    s.Org_Lvl_Nm,
    pt.pev_portfolioresponsiblecode,
    po.Annual_Premium,
    pt.Agrm_Sts_Cd,
    pt.Cncl_Dt,
    ROW_NUMBER() OVER (PARTITION BY po.Key_Policy ORDER BY po.Sales_Date
DESC) AS RowNum,
    CASE
      WHEN po.Annual_Premium >= 1500 THEN 'Premium'
      WHEN DATEDIFF(day, po.Vld_Fm_Tms, po.Vld_To_Tms) > 365 THEN 'Long-Term'
      ELSE 'Standard'
```

```
        END AS Policy_Type,
        CASE
            WHEN po.Annual_Premium >= 1000 THEN 'Good'
            ELSE 'Bad'
        END AS Agent_Performance
    FROM dbo.Portfolio_clean po
    JOIN dbo.Sales_Org_clean s ON po.Key_SS_Org = s.Key_SS_Org
    JOIN PolicyTransactions pt ON po.Key_Policy = pt.Key_Policy
    JOIN ConsistentPortfolioCodes cpc ON pt.Key_Policy = cpc.Key_Policy
    WHERE
        po.Payment_Status = 'Paid'
        AND (
            (s.Org_Lvl_Nm = 'Outbound' AND pt.pev_portfolioresponsiblecode = 'WILDWEST-3')
            OR (s.Org_Lvl_Nm = 'Internet' AND pt.pev_portfolioresponsiblecode = 'WILDWEST-3')
            OR (s.Org_Lvl_Nm = 'Inbound' AND pt.pev_portfolioresponsiblecode = 'WILDWEST-2')
        )
        AND s.fld_rep_cd NOT LIKE 'Inactive%'
),

-- Step 10: Determine the season for cancellation data
CancellationData AS (
    SELECT
        sp.Key_Policy,
        CASE
            WHEN MONTH(sp.Sales_Date) IN (12, 1, 2) THEN 'Q1'
            WHEN MONTH(sp.Sales_Date) IN (3, 4, 5) THEN 'Q2'
            WHEN MONTH(sp.Sales_Date) IN (6, 7, 8) THEN 'Q3'
            WHEN MONTH(sp.Sales_Date) IN (9, 10, 11) THEN 'Q4'
        END AS Season
    FROM SalesPortfolio sp
    GROUP BY sp.Key_Policy, sp.Sales_Date
```

```
)

-- Step 11: Final selection with calculated values (e.g., Commission, Season, Policy_Status)
SELECT
    sp.Key_Policy,
    sp.Org_Lvl_Nm,
    sp.Annual_Premium,
    sp.Product_Name,
    sp.pev_portfolioresponsiblecode,
    sp.Policy_Duration,
    sp.Payment_Status,
    sp.Sales_Date,
    -- Updated Commission_Amount calculation
    CASE
        WHEN sp.Sales_Date BETWEEN '2024-01-01' AND '2024-01-31' THEN 0 -- Commission is 0 for January 2024
        WHEN ROW_NUMBER() OVER (ORDER BY sp.Key_Policy) <= 1500 THEN 0.12 * sp.Annual_Premium
        ELSE 0.14 * sp.Annual_Premium
    END AS Commission_Amount,
    cd.Season
FROM SalesPortfolio sp
LEFT JOIN CancellationData cd ON sp.Key_Policy = cd.Key_Policy
WHERE sp.RowNum = 1;

-- Step 12: Update the 'EMD' table with calculated values such as Policy_Status, Commission_Eligibility, and Clawback
UPDATE EMD
SET
    Policy_Status = CASE WHEN sp.Policy_Duration >= 365 THEN 'Active' ELSE 'Cancel' END,
    Policy_Duration = COALESCE(sp.Policy_Duration, 0),
    Commission_Eligibility = CASE
```

```
    WHEN sp.Sales_Date BETWEEN '2024-01-01' AND '2024-01-31' THEN 'NO'
    ELSE
      CASE
        WHEN sp.pev_portfolioresponsiblecode IN ('WILDWEST-2', 'WILDWEST-3')
          AND (sp.Vld_Fm_Tms <= EMD.Sales_Date AND (sp.Vld_To_Tms IS NULL OR
sp.Vld_To_Tms >= EMD.Sales_Date))
          AND sp.Payment_Status = 'Paid'
          AND (sp.Product_Name BETWEEN 'Product 1' AND 'Product 8' OR
sp.Product_Name BETWEEN 'Product 13' AND 'Product 31')
          AND (
            (sp.pev_portfolioresponsiblecode = 'WILDWEST-3' AND (sp.Org_Lvl_Nm =
'Outbound' OR sp.Org_Lvl_Nm = 'Internet'))
            OR (sp.pev_portfolioresponsiblecode = 'WILDWEST-2' AND sp.Org_Lvl_Nm =
'Inbound')
          )
        THEN 'YES'
        ELSE 'NO'
      END,
  Cncl_Dt = CASE WHEN Policy_Status = 'Cancel' THEN sp.Cncl_Dt ELSE NULL END,
  -- Updated Clawback calculation
  Clawback = CASE
    WHEN sp.Sales_Date BETWEEN '2024-01-01' AND '2024-01-31' THEN 0 -- Clawback is
0 for January 2024
    WHEN Policy_Status = 'Cancel'
      AND sp.Cncl_Dt IS NOT NULL
      AND DATEDIFF(day, sp.Vld_Fm_Tms, sp.Vld_To_Tms) > 0
      AND (EMD.Sales_Date < '2024-01-01' OR EMD.Sales_Date > '2024-01-31')
    THEN
      ABS(
        (CAST(DATEDIFF(day, sp.Vld_Fm_Tms, sp.Cncl_Dt) AS FLOAT) /
         CAST(DATEDIFF(day, sp.Vld_Fm_Tms, sp.Vld_To_Tms) AS FLOAT)) *
sp.Annual_Premium
      )
```

```
        ELSE 0
    END,
    EMD.Season = cd.Season,
    EMD.Seasonal_Performance = CASE
        WHEN cd.Cancellation_Rate < 10 THEN 'Good'
        WHEN cd.Cancellation_Rate BETWEEN 10 AND 20 THEN 'Average'
        ELSE 'Poor'
    END
FROM SalesPortfolio sp
LEFT JOIN CancellationData cd ON sp.Key_Policy = cd.Key_Policy
WHERE sp.RowNum = 1;
```

CODE FOR AGGREGATE MONTHLY ANALYSIS :

```
-- Create a new table to store the summarized data with Commission Eligibility for the
Wild West dataset
CREATE TABLE [Wild West].dbo.Policy_Summary_Updated (
    Month VARCHAR(7),  -- Month-Year format
    Total_Policies INT,
    Active_Policies INT,
    Cancelled_Policies INT,
    New_Policies INT,
    Renewed_Policies INT,
    Premium_Revenue DECIMAL(18,2),
    Commission_Paid DECIMAL(18,2),
    Total_Clawback DECIMAL(18,2),
    Premium_Policies INT,
    Standard_Policies INT,
    Long_Term_Policies INT,
    Outbound_Sales INT,
    Internet_Sales INT,
    Inbound_Sales INT,
    Good_Performance_Agents INT,
    Poor_Performance_Agents INT,
    Seasonal_Performance VARCHAR(10),
    Commission_Eligibility VARCHAR(10)  -- New column for Commission Eligibility
);
SELECT
    FORMAT(Sales_Date, 'yyyy-MM') AS Month,  -- Month-Year format
    COUNT(Key_Policy) AS Total_Policies,
    SUM(CASE WHEN Policy_Status = 'Active' THEN 1 ELSE 0 END) AS Active_Policies,
    SUM(CASE WHEN Policy_Status = 'Cancel' THEN 1 ELSE 0 END) AS Cancelled_Policies,
    SUM(CASE WHEN Policy_Type = 'New' THEN 1 ELSE 0 END) AS New_Policies,
    SUM(CASE WHEN Policy_Type = 'Renewed' THEN 1 ELSE 0 END) AS Renewed_Policies,
    SUM(Annual_Premium) AS Premium_Revenue,
    SUM(Commission) AS Commission_Paid,
    SUM(Clawback) AS Total_Clawback,  -- Total Clawback
    SUM(CASE WHEN Policy_Type = 'Premium' THEN 1 ELSE 0 END) AS Premium_Policies,
    SUM(CASE WHEN Policy_Type = 'Standard' THEN 1 ELSE 0 END) AS Standard_Policies,
    SUM(CASE WHEN Policy_Type = 'Long-Term' THEN 1 ELSE 0 END) AS Long_Term_Policies,
    SUM(CASE WHEN Org_Lvl_Nm = 'Outbound' THEN 1 ELSE 0 END) AS Outbound_Sales,
    SUM(CASE WHEN Org_Lvl_Nm = 'Internet' THEN 1 ELSE 0 END) AS Internet_Sales,
    SUM(CASE WHEN Org_Lvl_Nm = 'Inbound' THEN 1 ELSE 0 END) AS Inbound_Sales,
    SUM(CASE WHEN Agent_Performance = 'Good' THEN 1 ELSE 0 END) AS
Good_Performance_Agents,
    SUM(CASE WHEN Agent_Performance = 'Bad' THEN 1 ELSE 0 END) AS
Poor_Performance_Agents,
    CASE
        WHEN AVG(Annual_Premium) >= 1500 THEN 'Good'
        WHEN AVG(Annual_Premium) BETWEEN 1000 AND 1499 THEN 'Average'
        ELSE 'Poor'
    END AS Seasonal_Performance
FROM dbo.Month_Data_8
WHERE Sales_Date BETWEEN '2024-01-01' AND '2024-08-31'
  AND Commission_Eligibility = 'YES'  -- Filter only rows with Commission Eligibility
= 'YES'
GROUP BY FORMAT(Sales_Date, 'yyyy-MM')
ORDER BY Month ASC;
```

CODE FOR EXTRA ANALYSIS : MONTHLY BREAKDOWN OF ACTIVE VS CANCELED POLICIES

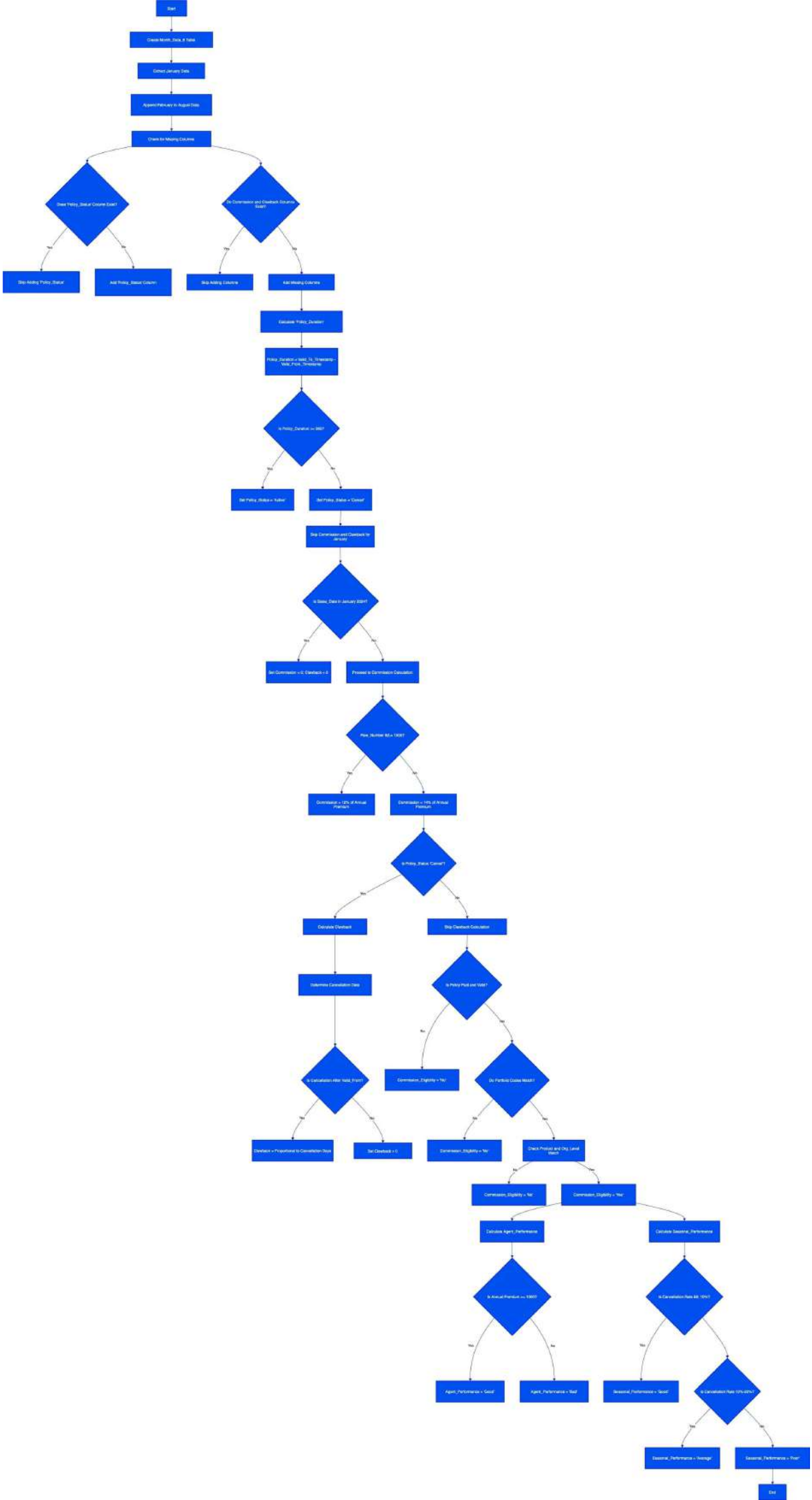
```
USE [Wild West]
-- Create a new table and insert the result of the query
SELECT
    EMD.Sales_Date,                             -- The specific date for the policy
    EMD.Seasonal_Performance,                   -- Seasonal performance (Good, Average,
Poor)
    EMD.Policy_Status,                          -- Policy status (Active/Cancel) on the
given date
    COUNT(*) AS PolicyCount,                    -- Count of policies for the specific
date
    SUM(CASE WHEN EMD.Policy_Status = 'Active' THEN 1 ELSE 0 END) AS ActivePolicies,
-- Count of Active policies
    SUM(CASE WHEN EMD.Policy_Status = 'Cancel' THEN 1 ELSE 0 END) AS CanceledPolicies,
-- Count of Canceled policies
    AVG(EMD.Cancellation_Rate) AS AvgCancellationRate   -- Average cancellation rate
for that specific date
INTO dbo.PolicySeasonalSummary   -- This creates the new table
FROM dbo.Month_Data_8 EMD
GROUP BY
    EMD.Sales_Date,                             -- Group by specific date
    EMD.Seasonal_Performance,                   -- Group by Seasonal Performance on
that specific date
    EMD.Policy_Status                            -- Group by Policy Status
(Active/Cancel) on that specific date
ORDER BY
    EMD.Sales_Date,                             -- Order results by Sales Date
    EMD.Seasonal_Performance,                   -- Then by Seasonal Performance (Good,
Average, Poor)
    EMD.Policy_Status;                          -- Finally, by Policy Statustus
(Active/Cancel)
```

CODE FOR THE DATA MODEL

```
use [Wild West]
CREATE TABLE Policy (
	Key_Policy INT PRIMARY KEY,
	Ext_Refr NVARCHAR(50),
	Term TINYINT,
	Vrsn TINYINT,
	Sub_Vrsn TINYINT,
	Agrm_Sts_Cd NVARCHAR(10),
	Prtn NVARCHAR(50)
);
CREATE TABLE PolicyDates (
	PolicyDate_ID INT PRIMARY KEY IDENTITY,
	Key_Policy INT REFERENCES Policy(Key_Policy),
	Incp_Dt DATE,
	Bnd_Dt DATE,
	Pln_End_Dt DATE,
	Rnew_Dt DATE,
	Cncl_Dt DATE,
	Sts_Dt DATE
);
CREATE TABLE SalesOrganization (
	Key_SS_Org INT,
	Key_Policy INT REFERENCES Policy(Key_Policy),
	Fld_Rep_Cd NVARCHAR(10),
	Org_Lvl_cd NVARCHAR(10),
	Org_Lvl_Nm NVARCHAR(50)
);
CREATE TABLE Portfolio (
	Portfolio_ID INT PRIMARY KEY IDENTITY,
	Key_Policy INT REFERENCES Policy(Key_Policy),
	Key_SS_Org INT,
	Sales_Date DATE,
	Annual_Premium BIGINT
);
CREATE TABLE PortfolioValidity (
	Validity_ID INT PRIMARY KEY IDENTITY,
	Portfolio_ID INT REFERENCES Portfolio(Portfolio_ID),
	Vld_Fm_Tms DATETIME2,
	Vld_To_Tms DATETIME2
);
CREATE TABLE PortfolioStatus (
	PortfolioStatus_ID INT PRIMARY KEY IDENTITY,
	Portfolio_ID INT REFERENCES Portfolio(Portfolio_ID),
	Payment_Status NVARCHAR(20),
	PaymentDate DATE
);
CREATE TABLE Product (
	Product_ID INT PRIMARY KEY IDENTITY,
	product_Name NVARCHAR(50)
);
CREATE TABLE PortfolioProduct (
	PortfolioProduct_ID INT PRIMARY KEY IDENTITY,
	Portfolio_ID INT REFERENCES Portfolio(Portfolio_ID),
	Product_ID INT REFERENCES Product(Product_ID),
	No_Of_Sold_Policies TINYINT,
	No_Of_Cancelled_Policies INT,
	No_Of_Paid_Policies INT
```

```
	);
	CREATE TABLE [Transaction] (
		Transaction_ID INT PRIMARY KEY IDENTITY,
		Created_At DATETIME2,
		Portfolio_Responsible_Code NVARCHAR(50),
		Key_Policy INT REFERENCES Policy(Key_Policy),
		Portfolio_No NVARCHAR(50)
	);
	CREATE TABLE CancellationPolicy (
		Key_Policy INT REFERENCES Policy(Key_Policy),
		Cancel_Before_Vld_Frm INT,
		Cancel_After_Vld_Frm INT
	);
	CREATE TABLE TransactionResponsible (
		Responsible_ID INT PRIMARY KEY IDENTITY,
		Transaction_ID INT REFERENCES [Transaction](Transaction_ID),
		Pev_PortfolioResponsibleCode NVARCHAR(50)
	);
	use [Wild West]
	-- Insert statements with updated column names
	INSERT INTO Policy (Key_Policy, Ext_Refr, Term, Vrsn, Sub_Vrsn, Agrm_Sts_Cd,
Prtn)
	SELECT Key_Policy, Ext_Refr, Term, Vrsn, Sub_Vrsn, Agrm_Sts_Cd, Prtn FROM
dbo.Policy_clean;
	INSERT INTO PolicyDates (Key_Policy, Incp_Dt, Bnd_Dt, Pln_End_Dt, Rnew_Dt,
Cncl_Dt, Sts_Dt)
	SELECT Key_Policy, Incp_Dt, Bnd_Dt, Pln_End_Dt, Rnew_Dt, Cncl_Dt, Sts_Dt FROM
dbo.Policy_clean;
	INSERT INTO SalesOrganization (Key_SS_Org, Key_Policy, Fld_Rep_Cd, Org_Lvl_cd,
Org_Lvl_Nm)
	SELECT DISTINCT Sales_Org_clean.Key_SS_Org, dbo.Policy_clean.Key_Policy,
Sales_Org_clean.Fld_Rep_Cd, Sales_Org_clean.Org_Lvl_cd, Sales_Org_clean.Org_Lvl_Nm
	FROM Sales_Org_clean
	INNER JOIN Portfolio_clean ON Sales_Org_clean.Key_SS_Org =
Portfolio_clean.Key_SS_Org
	INNER JOIN Policy_clean ON Portfolio_clean.Key_Policy =
Policy_clean.Key_Policy;
	INSERT INTO Portfolio (Key_Policy, Key_SS_Org, Sales_Date, Annual_Premium)
	SELECT lp.Key_Policy, lp.Key_SS_Org, lp.Sales_Date, lp.Annual_Premium FROM
dbo.Portfolio_clean lp;
	INSERT INTO PortfolioValidity (Portfolio_ID, Vld_Fm_Tms, Vld_To_Tms)
	SELECT p.Portfolio_ID, lp.Vld_Fm_Tms, lp.Vld_To_Tms
	FROM dbo.Portfolio_clean lp
	INNER JOIN Portfolio p ON lp.Key_Policy = p.Key_Policy;
	INSERT INTO PortfolioStatus (Portfolio_ID, Payment_Status, PaymentDate)
	SELECT p.Portfolio_ID, lp.Payment_Status, lp.PaymentDate
	FROM dbo.Portfolio_clean lp
	INNER JOIN Portfolio p ON lp.Key_Policy = p.Key_Policy;
	INSERT INTO Product (product_Name)
	SELECT DISTINCT product_Name FROM dbo.Portfolio_clean WHERE product_Name IS NOT
NULL;
	INSERT INTO PortfolioProduct (Portfolio_ID, Product_ID, No_Of_Sold_Policies,
No_Of_Cancelled_Policies, No_Of_Paid_Policies)
	SELECT p.Portfolio_ID, pr.Product_ID, lp.No_Of_Sold_Policies,
lp.No_Of_Cancelled_Policies, lp.No_Of_Paid_Policies
	FROM dbo.Portfolio_clean lp
	INNER JOIN Portfolio p ON lp.Key_Policy = p.Key_Policy
	INNER JOIN Product pr ON lp.product_Name = pr.product_Name;
	INSERT INTO [Transaction] (Created_At, Portfolio_Responsible_Code, Key_Policy,
Portfolio_No)
	SELECT pev_CreatedAt, Pev_PortfolioResponsibleCode, Key_Policy, po_No FROM
dbo.Policy_Transactions_clean;
```

```
	INSERT INTO CancellationPolicy (Key_Policy, Cancel_Before_Vld_Frm,
Cancel_After_Vld_Frm)
	SELECT Key_Policy, Cancel_Before_Vld_Frm, Cancel_After_Vld_Frm FROM
dbo.Portfolio_clean;
	INSERT INTO TransactionResponsible (Transaction_ID,
Pev_PortfolioResponsibleCode)
	SELECT t.Transaction_ID, lt.Pev_PortfolioResponsibleCode
	FROM Policy_Transactions_clean AS lt
	INNER JOIN [Transaction] AS t ON lt.Key_Policy = t.Key_Policy;
```


Start
Create Month_Data_It Table
Collect January Data
Append February to August Data
Check for Missing Columns
Does Policy_Status' Column Exist?
Do Commission and Clawback Columns Exist?
Skip Adding 'Policy_Status'
Add 'Policy_Status' Column
Skip Adding Columns
Add Missing Columns
Calculate 'Policy_Duration'
Policy_Duration = Valid_To_Timestamp - Valid_From_Timestamp
Is Policy_Duration >= 365?
Set Policy_Status = 'Active'
Set Policy_Status = 'Closed'
Skip Commission and Clawback for January
Is Sales_Data in January 2024?
Set Commission = 0, Clawback = 0
Proceed to Commission Calculation
Plan_Number 60 or 190?
Commission = 12% of Annual Premium
Commission = 15% of Annual Premium
Is Policy_Status 'Canceled'?
Calculate Clawback
Skip Clawback Calculation
Determine Cancellation Date
Is Policy Paid and Valid?
Is Calculation After Valid_From?
Clawback = Proportional to Cancellation Days
Set Clawback = 0
Commission_Eligibility = 'No'
Do Portfolio Codes Match?
Commission_Eligibility = 'No'
Check Product and Org. Level Match
Commission_Eligibility = 'No'
Commission_Eligibility = 'Yes'
Calculate Agent_Performance
Calculate Seasonal_Performance
Is Annual Premium >= 15000?
Is Cancellation Rate 10% or more?
Agent_Performance = 'Good'
Agent_Performance = 'Bad'
Seasonal_Performance = 'Good'
Is Cancellation Rate 10%-20%?
Seasonal_Performance = 'Average'
Seasonal_Performance = 'Poor'
End