Insurance Data Analytics
Power BI Project Documentation

Data source: InsuranceData (CSV) imported into SQL Server
Last Updated: January 17, 2026

Author: Mgr. Tamás Csiba

Contents

1. Introduction

This Power BI project analyzes an insurance portfolio by combining policy-level information with claim outcomes to produce interactive insights. The report is designed to support day-to-day monitoring (premium volume, coverage exposure, and claim status distribution) and exploratory analysis (claim patterns across policy types, age groups, and customer segments).

As a first step, the raw dataset was loaded from a CSV file into Microsoft SQL Server to create a structured, reusable source table for Power BI modeling, DAX measures, and dashboard development.

2. Data Source and SQL Server Import

2.1 Data Source

The project is based on a CSV file named InsuranceData, which contains records related to insurance policies and claims.

2.2 Import into SQL Server

The CSV file was imported into Microsoft SQL Server and loaded into:

Database: Insurancedb

Schema/Table: dbo.InsuranceData

Row count: 10,004 rows (confirmed after import)

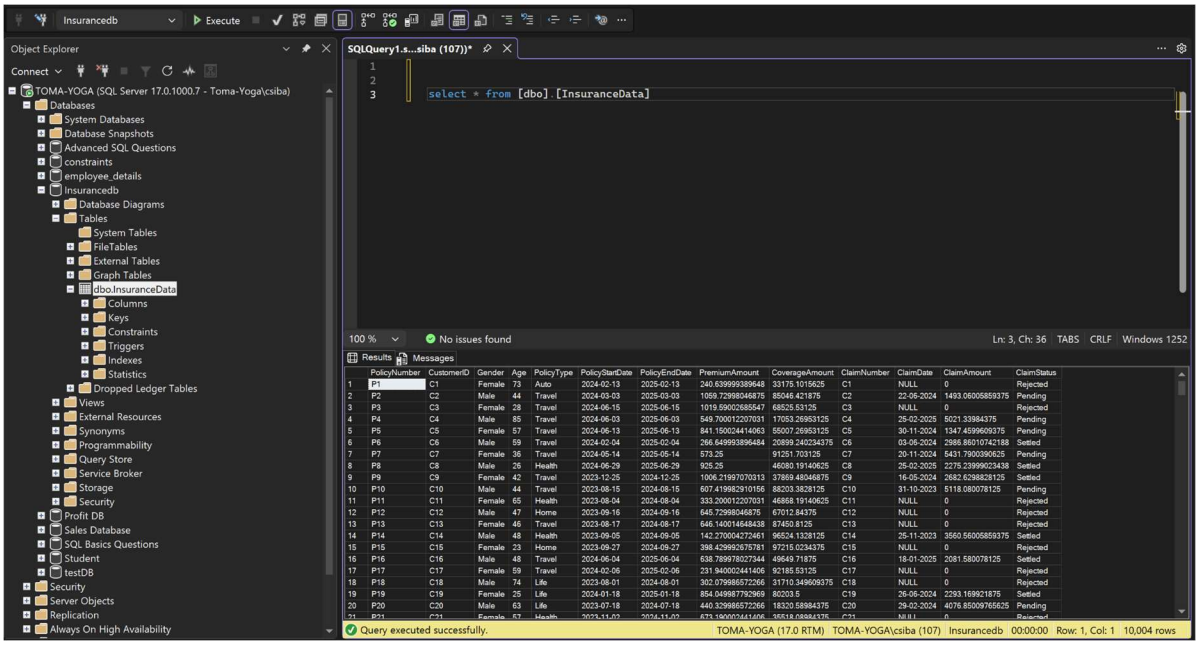

Figure 1 - InsuranceData successfully imported into SQL Server (Insurancedb.dbo.InsuranceData).

2.3 Validation Query

After loading the table, a direct query was executed to confirm that the data is accessible and returns the expected records:

```
SELECT *
FROM [dbo].[InsuranceData];
```

2.4 Main Columns (High Level)

The dbo.InsuranceData table includes the key attributes required for reporting and analysis:

PolicyNumber - Unique policy identifier

CustomerID - Customer identifier

Gender - Customer gender

Age - Customer age

PolicyType - Insurance product category (e.g., Auto, Travel, Health, Home, Life)

PolicyStartDate - Policy start date

PolicyEndDate - Policy end date

PremiumAmount - Premium value associated with the policy

CoverageAmount - Coverage/insured amount (exposure)

ClaimNumber - Claim identifier (when applicable)

ClaimDate - Date of the claim event (when applicable)

ClaimAmount - Monetary value of the claim (when applicable)

ClaimStatus - Claim processing status (e.g., Pending / Settled / Rejected)

2.5 Data Quality Notes

Claim-related fields can contain NULL values (commonly ClaimDate and ClaimAmount) when a policy has no associated claim.

Date fields will be typed and validated as proper dates to support filtering and time intelligence.

Monetary columns (PremiumAmount, CoverageAmount, ClaimAmount) will be treated as decimal/currency to ensure correct aggregation.

3. Power BI - SQL Server Connection

To build the report on a reliable and reusable data source, the Insurance dataset was connected to Power BI directly from Microsoft SQL Server. The connection was created using the native 'SQL Server database' connector. The server name used was 'TOMA-YOGA' and the data was imported in Import mode, enabling fast in-memory analytics, DAX calculations, and interactive visuals.

Connection details (high level):

- Server: TOMA-YOGA
- Connectivity mode: Import
- Source table: Insurancedb.dbo.InsuranceData

Figure 2 - Power BI SQL Server connector settings used for importing the dataset.

4. Data Preparation (Power Query)

To support segmentation and clearer reporting, two additional derived columns were created in Power Query using Conditional Column rules. These columns are used for filtering, grouping, and high-level analysis in the report.

4.1 Age Group (Conditional Column)

An 'Age Group' column was created from the 'Age' field to enable demographic segmentation in visuals and slicers. The following business rules were applied:

- Young Adults: Age <= 24
- Adult: Age <= 60
- Elder: Age > 60

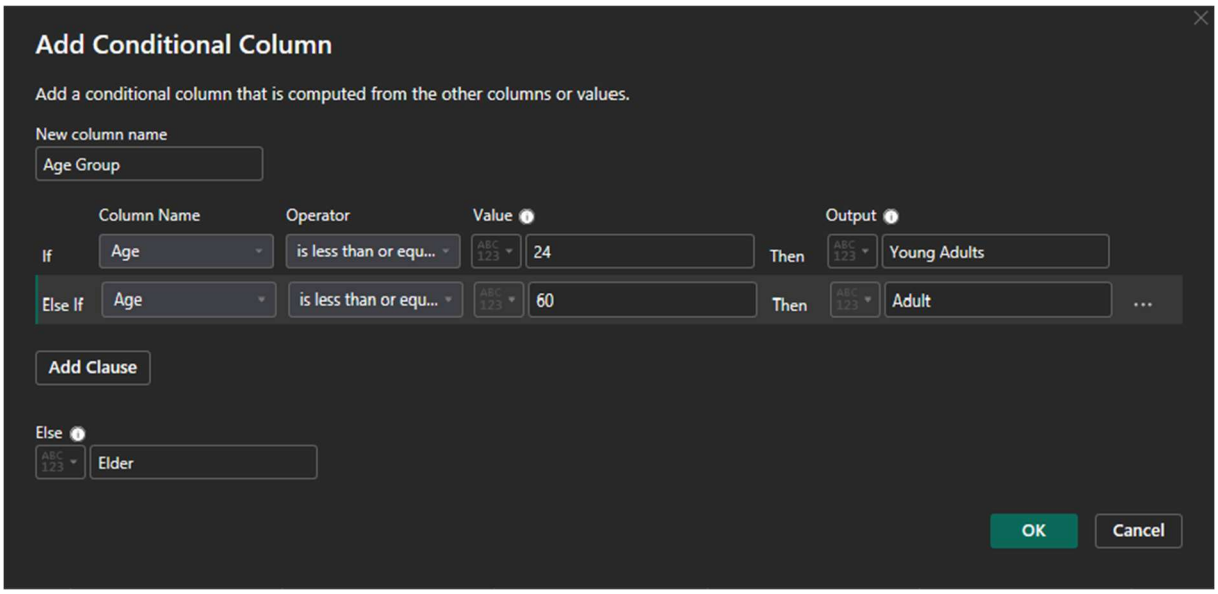

Figure 3 - Power Query Conditional Column: Age Group derivation.

4.2 Active vs Inactive Policy (Conditional Column)

An 'Active/Inactive' status column was created based on 'PolicyEndDate' to differentiate active policies from expired ones. A cut-off date was used to classify policies:

- Inactive: PolicyEndDate <= 2024-12-10
- Active: otherwise

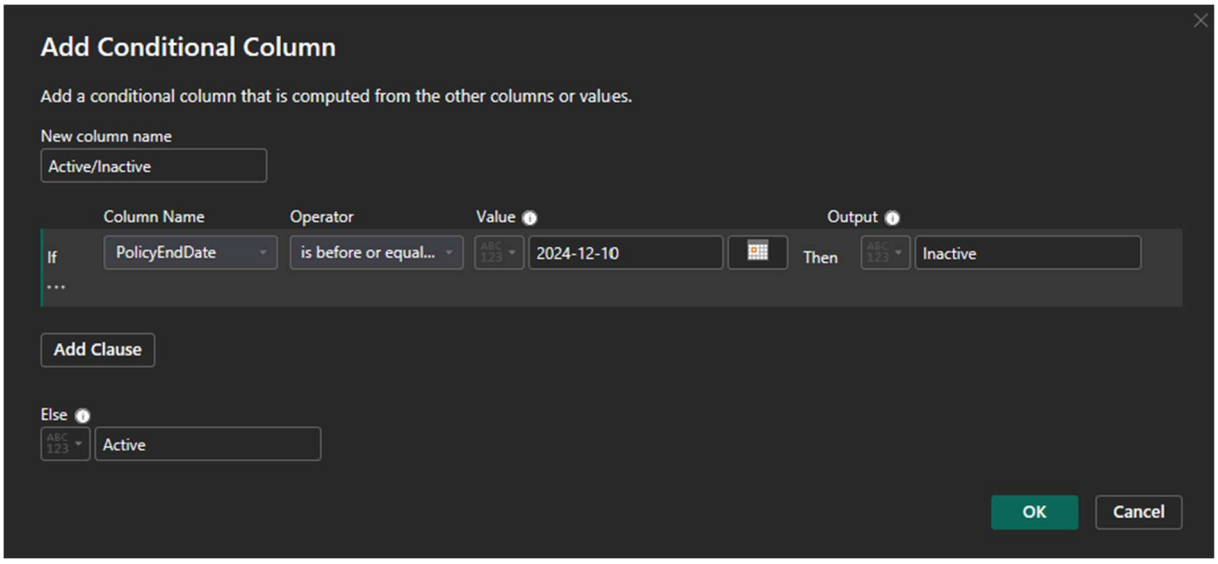

Figure 4 - Power Query Conditional Column: Active vs Inactive policy status.

5. Report Pages (Report View)

This section documents the Power BI report view pages and explains the purpose of each visual, the interactive filtering behavior, and the business questions supported by the dashboard.

5.1 Overview Dashboard

The Overview Dashboard provides a single-page summary of the insurance portfolio, combining policy-level exposure (premium and coverage) with claim outcomes to support fast monitoring and exploratory analysis.

Key objectives

- Provide an executive snapshot using KPI cards (Premium Amount, Coverage Amount, Claim Amount).
- Enable quick filtering by PolicyNumber, ClaimNumber, and CustomerID for case-level analysis.
- Highlight portfolio composition and performance by policy lifecycle status, product type, demographics, and claim status.

Visuals and interactions

- Slicers: PolicyNumber, ClaimNumber, CustomerID - filter the entire page context.
- KPI Cards: Total Premium Amount, Total Coverage Amount, Total Claim Amount - update dynamically based on filters and selections.
- Donut Chart: Count of Active/Inactive Policies - summarizes policy lifecycle status using the derived Active/Inactive column.
- Bar Chart: Premium Amount by Policy Type - ranks product categories by premium volume.
- Column Chart: Number of Claims by Claim Status - shows the distribution of Rejected, Settled, and Pending claims.
- Line/Area Chart: Claim Amount by Age Group - compares claim cost across Young Adults, Adult, and Elder segments.
- Matrix: Amounts by Policy Type and Claim Status - detailed breakdown to validate KPI totals and identify concentration areas.

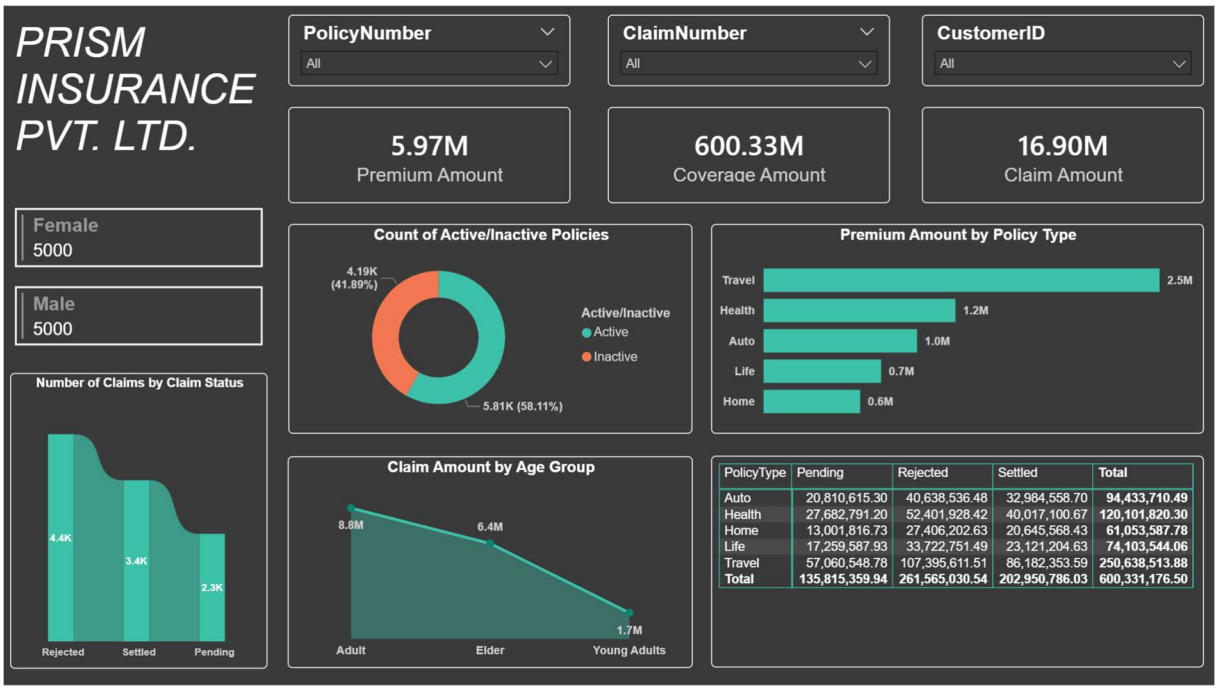

Figure 5 - Power BI report view: Overview dashboard summarizing portfolio KPIs and breakdowns.

5.2 Drill Through Table

This page is configured as a drill-through target to support record-level investigation behind the summary visuals. It displays the underlying rows in a table format and applies the drill-through filter context from the originating visual.

Drill-through configuration:

- Drill-through field: PolicyType
- The page receives the selected PolicyType value (e.g., Travel) and filters the detail table accordingly.
- This enables validation of totals and quick identification of outliers (e.g., high claims, rejected claims, unusual coverage or premium values).

The table includes key policy and claim attributes such as PolicyNumber, CustomerID, ClaimNumber, Age, Gender, CoverageAmount, PremiumAmount, PolicyStartDate, PolicyEndDate, ClaimStatus, and ClaimDate.

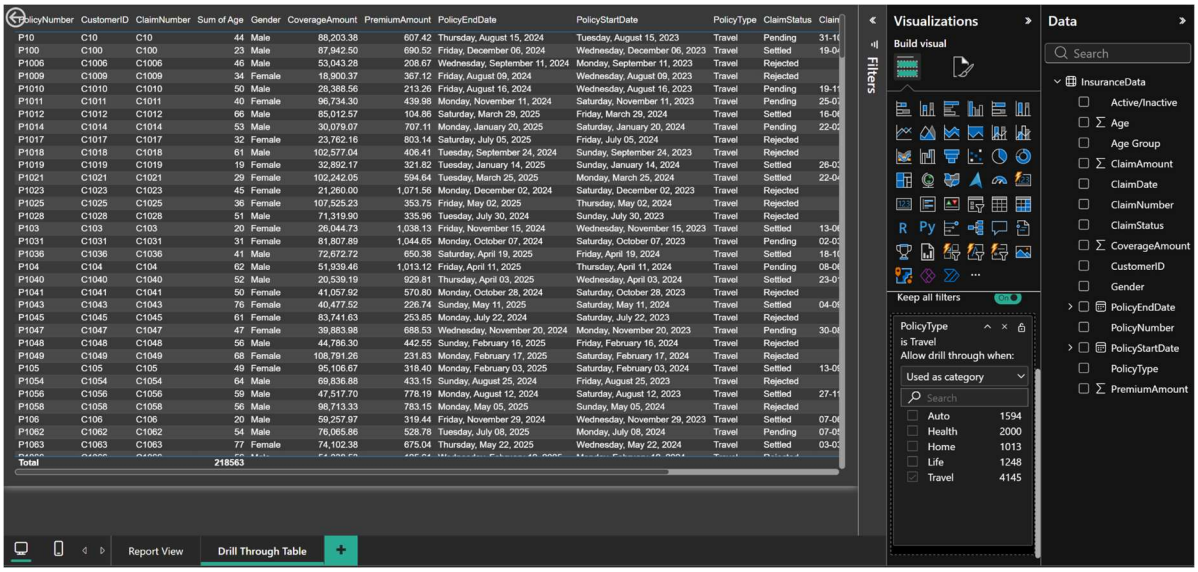

Figure 6 - Drill Through Table page filtered by PolicyType to show record-level policy and claim details.

6. Row-Level Security (RLS)

To demonstrate secure data access in Power BI, Row-Level Security (RLS) was implemented using role-based filters. RLS ensures that different user groups can only see the subset of records they are authorized to access, while using the same report and data model.

6.1 Security roles based on PolicyType

Two sample roles were created in Power BI Desktop under Manage roles. Each role restricts rows in the InsuranceData table by filtering the PolicyType column:

- Health Role: PolicyType equals "Health"
- Travel Role: PolicyType equals "Travel"

With these roles in place, the same visuals and measures automatically recalculate based on the role filter, providing a realistic example of how department- or product-specific access can be enforced in a shared reporting solution.

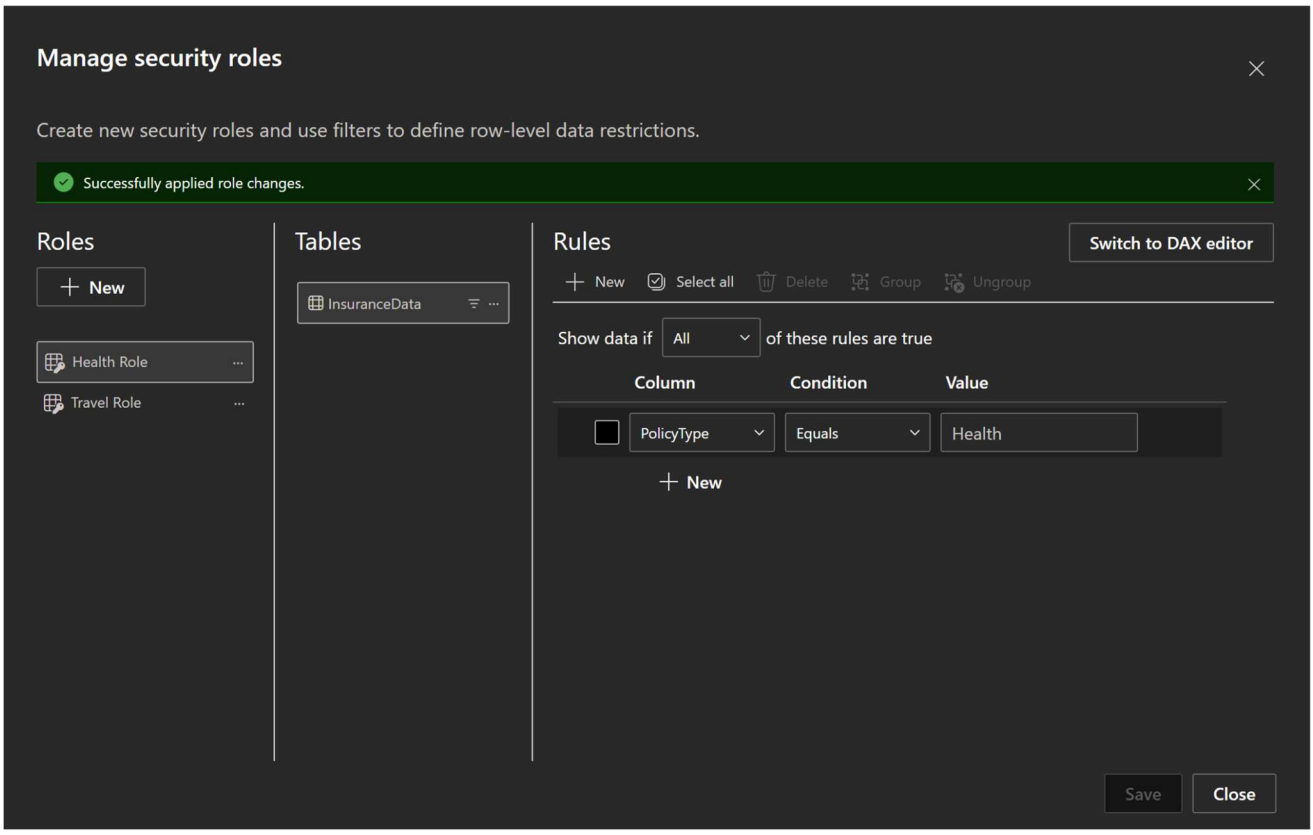

Figure 7 - Power BI Manage roles: example RLS roles filtering InsuranceData by PolicyType.

Summary

This project delivers an end-to-end **Power BI insurance analytics report** built on a structured SQL Server data source. The workflow starts with loading the InsuranceData CSV into **SQL Server** (Insurancedb.dbo.InsuranceData, 10,004 rows), providing a stable, reusable backend for reporting. The dataset includes essential policy, customer, and claim attributes (premium, coverage, claim status, dates, demographics), enabling both portfolio monitoring and deeper claim analysis.

In Power BI, the data is connected via the **SQL Server connector in Import mode**, ensuring fast in-memory performance for interactive visuals and DAX calculations. During preparation, two additional derived columns were created in **Power Query** to support segmentation and clearer reporting: **Age Group** (Young Adults / Adult / Elder) and **Active/Inactive** policy status based on PolicyEndDate. The report's main **Overview dashboard** presents key KPIs (Premium, Coverage, Claim Amount) and multiple analytical breakdowns (Active vs Inactive distribution, premium by policy type, claim status counts, claim amount by age group, and a detailed matrix). For detailed investigation, a **Drill Through Table** page was implemented using PolicyType as the drill-through field, enabling users to trace summary insights back to individual records. Finally, **Row-Level Security (RLS)** was configured with sample roles (e.g., Health Role, Travel Role) to demonstrate secure, role-based access control within the same report.