

Data Analytics Microsoft Power Bi HR Project

Submitted By

Team 3 - CLS CAI3_DAT2_G4

21108707	Abanoub Aziz Fawzy
21106815	Rania Abdelfatah Mansour Ali
21108002	Reham Ragab Mohamed Ali
21108600	Rasha Ragab Mohamed Ali
21042702	Mohammed Morsy El Sayed Moustafa
21100277	Ola Mohamed Mahmoud Mohamed Kadry

Contents

1. Introduction	3
2. Task Division	3
3. Research Question, Problem Definition	5
4. Power Query Transformation Documentation.....	8
5. Power BI Data Model	12
6.Metric Group: Attrition	15
7. Data Coverage Metrics	17
8. Salary Insights Metrics.....	18
9. Managerial Tenure Insights Metrics	20
10. Calculated columns in DimEmployee Table	21
11. FactPerformanceRating – Calculated Columns Documentation	28
12. Performance Flags Metrics (FactPerformanceRating)	29
13. Project results and outcome.....	33
14. Proposed solutions	33
15.Conclusion	34

1. Introduction

Human Resource (HR) data contains important information about employees, but many organizations do not use it effectively. This project focuses on turning raw HR data into useful insights that can help managers make better decisions. By analyzing factors such as employee satisfaction, performance, salary, age, and work-life balance, we aim to understand what influences employee behavior—especially why some employees leave the company.

Using Power BI, we cleaned the data, built a structured model, and created visual dashboards that answer key HR questions. These dashboards help identify trends, problem areas, and opportunities for improvement. Overall, the project shows how data analytics can help organizations reduce attrition, improve employee satisfaction, and create a healthier and more productive work environment.

1. Task Division

Objective: To divide the project work into adequate sub-tasks, distribute them equitably among the 5 group members, and ensure timely reporting.

Member 1: Project Lead & Data Engineer

- **Primary responsibility for Week 1: Data Cleaning and Preprocessing.**
 1. **Data Integration:** Connect and integrate all provided data tables within Power BI's Power Query editor.
 2. **Data Cleansing:** Clean the raw data by handling missing values, standardizing formats (e.g., HireDate), and correcting data types to ensure accuracy and consistency.
 3. **Data Modeling:** Establish and manage relationships between tables (e.g., linking EmployeeID across fact and dimension tables) to build a robust and efficient data model.
 4. **Feature Engineering:** Create necessary calculated columns, measures, and hierarchies to support the planned analysis and dashboard visualizations.
 5. **Quality Assurance:** Validate the dataset to ensure it is well-structured, accurate, and fully prepared for in-depth analysis.

6. **Reporting & Documentation:** Deliver the finalized, cleaned dataset and comprehensive "Data Preprocessing Documentation," while proactively reporting on the data model's readiness and any challenges encountered.

Member 2: Data Analyst

- **Primary responsibility for Week 2: Analysis Questions Phase.**
 1. **Question Ideation:** Brainstorm a comprehensive list of potential business questions across key HR domains (e.g., Attrition, Satisfaction, Compensation, Diversity, Performance).
 2. **Question Prioritization:** Evaluate and prioritize the questions based on their potential value and impact for HR decision-makers.
 3. **Technical Validation:** Collaborate with the Data Engineer to ensure the cleaned data model can support the chosen questions.
 4. **Hypothesis Testing:** Conduct exploratory data analysis (EDA) within Power BI to preliminarily test the hypotheses underlying the key questions.
 5. **Reporting:** Deliver the final "Set of analysis questions" and explaining the rationale behind their selection.

Member 3 & 4: Dashboard Developers (Visualization Specialists)

- **Primary responsibility for Week 3: Build Dashboard.**
 1. **Dashboard Design:** Architect the overall dashboard layout, prioritizing clarity, visual impact, and user-friendliness for the end-user.
 2. **Visualization Development:** Build interactive and insightful visualizations that directly answer the defined research questions.
 3. **Interactivity Implementation:** Incorporate filters, slicers, and tooltips to create a dynamic and user-driven analytical experience.
 4. **Aesthetic Standardization:** Apply and maintain a consistent visual theme, including formatting, color palette, and branding throughout the dashboard.
 5. **Reporting & Demonstration:** Responsible for presenting the functional dashboard and articulating the rationale behind its design and functionality.

Member 5 & 6: Final Report and Presentation Coordinator

- **Primary responsibility for Week 4: Final Presentation and Report.**

1. **Process Documentation:** Meticulously document the project's methodology, key challenges encountered, and the analytical process.
2. **Insight Synthesis:** Analyze and summarize the key findings and actionable insights derived from the data analysis.
3. **Report Compilation:** Author the final written report, incorporating inputs from all team members and organizing the results in a clear and logical manner.
4. **Presentation Development:** Design and create the final presentation slides that effectively summarize the project's work and conclusions.
5. **Project Coordination:** Responsible for the final deliverables (report and presentation) and for coordinating the team's rehearsal for the final presentation.

Overall Team Collaboration: Final Report and Presentation Coordinator

1. While primary development of the report and presentation was assigned to a designated coordinator, all team members collaboratively presented the final deliverable.
2. All members actively participated in the Q&A session, ensuring a comprehensive and unified response to questions.
3. The division of labor focused on primary responsibilities; however, all members participated in brainstorming sessions for key steps—including question formulation, dashboard design, and analysis interpretation—to ensure full team alignment and input throughout the project lifecycle.

3. Research Question, Problem Definition

Problem Definition

- Human Resources departments hold a wealth of data on their employees, but this data is often underutilized. Without transforming this data into actionable insights, organizations risk increased employee turnover, lower productivity, poor employee morale, and inefficient allocation of resources like training and compensation. This project aims to leverage data analytics to solve core HR challenges by identifying key factors influencing employee attrition, satisfaction,

and performance, thereby empowering decision-makers to create targeted strategies for improving employee retention and overall organizational health.

- **The problem addressed in this project is:**

- How can employee data (demographics, performance ratings, satisfaction levels, and compensation) be analyzed to provide actionable insights for reducing attrition, improving employee satisfaction, and enhancing decision-making in human resource management?
- The insights derived from this analysis are directly significant to organizational decision-makers (HR Managers, Department Heads, Executives) by:
 - **Reducing Costs:** High employee turnover is extremely costly due to recruitment, onboarding, and lost productivity.
 - **Improving Strategy:** Informing data-driven policies on compensation, benefits, training programs, and hiring practices.
 - **Enhancing Culture:** Identifying drivers of job satisfaction and work-life balance to improve overall employee experience.
 - **Boosting Performance:** Understanding the link between satisfaction, training, and performance to optimize workforce output.

Research Questions

1. Attrition Analysis:

- What are the key factors that most strongly predict employee attrition?
- Is there a significant difference in attrition rates between different departments, job roles, or marital statuses?
- How do factors like DistanceFromHome, Overtime, Salary, and StockOptionLevel influence an employee's decision to leave?
- Does business travel frequency impact employee satisfaction or attrition?
- What patterns exist between performance ratings and employee attrition?
- Are marital status and gender linked to different levels of attrition or satisfaction?
- Is there a correlation between education level and job role with attrition?

- Which factors (demographics, compensation, performance) are the strongest predictors of attrition?

2. Satisfaction & Performance:

- What is the relationship between JobSatisfaction/EnvironmentSatisfaction and JobRole or Department?
- Does a higher WorkLifeBalance score correlate with higher JobSatisfaction or lower Attrition?
- Is there a gap between SelfRating and ManagerRating? Does the size of this gap correlate with JobSatisfaction?
- What is the relationship between employees' age and their satisfaction levels?
- How does work-life balance vary across departments and age groups?

3. Compensation & Benefits:

- How is Salary distributed across different Departments, JobRoles, and Genders? Are there any noticeable disparities?
- Is there a correlation between YearsAtCompany/YearsSinceLastPromotion and Salary? (Pay progression analysis)
- Does having OverTime significantly impact an employee's Salary or StockOptionLevel?

4. Diversity & Inclusion:

- What is the demographic composition (Gender, Ethnicity, Age) of the company across different departments and job roles?
- Is there a relationship between Gender or Ethnicity and Salary, Promotion (YearsSinceLastPromotion), or PerformanceRating?

5. Career Development:

- What is the typical career path (YearsInMostRecentRole, YearsSinceLastPromotion) within the company?

- Do employees who take more TrainingOpportunitiesTaken receive promotions faster or have higher performance ratings?
- How does EducationField align with current JobRole and does this alignment impact JobSatisfaction?
- How do training opportunities relate to employees' performance ratings and satisfaction?

6. General HR Health:

- What is the overall tenure and age distribution of our workforce?
- How many employees are at risk of burnout (high Overtime, low WorkLifeBalance)?
- Which managers (YearsWithCurrManager) have teams with higher satisfaction and lower attrition rates?

4. Power Query Transformation Documentation

1-Employee Table

Transformations Applied

1. Source Table: Employee (renamed to DimEmployee)
2. Trimmed all text columns – removed leading/trailing spaces.
3. Merged FirstName and LastName into FullName – separated by a space; original columns removed.
4. Converted OverTime and Attrition to Boolean – Yes → true, No → false.
5. Validated all other columns – ensured correct data types (e.g., numeric, date, categorical).
6. Education column → renamed to EducationLevelID.

Column Name	Data Type	Key Type
EmployeeID	Text	Primary Key
FullName	Text	—
Gender	Text	—
Age	Whole Number	—

Column Name	Data Type	Key Type
BusinessTravel	Text	—
Department	Text	—
DistanceFromHome (KM)	Whole Number	—
State	Text	—
Ethnicity	Text	—
EducationLevelID	Whole Number	Foreign Key
EducationField	Text	—
JobRole	Text	—
MaritalStatus	Text	—
Salary	Fixed Decimal	—
StockOptionLevel	Whole Number	—
OverTime	Boolean	—
HireDate	Date	—
Attrition	Boolean	—
YearsAtCompany	Whole Number	—
YearsInMostRecentRole	Whole Number	—
YearsSinceLastPromotion	Whole Number	—
YearsWithCurrManager	Whole Number	—

2- PerformanceRating Table

Transformations Applied

1. Source Table: PerformanceRating (renamed to FactPerformanceRating)
2. Ensured all columns are in the correct data type.

Column Name	Data Type	Key Type
PerformanceID	Text	Primary Key
EmployeeID	Text	Foreign Key
ReviewDate	Date	—
EnvironmentSatisfaction	Whole Number	Foreign Key
JobSatisfaction	Whole Number	Foreign Key
RelationshipSatisfaction	Whole Number	Foreign Key
TrainingOpportunitiesWithinYear	Whole Number	—
TrainingOpportunitiesTaken	Whole Number	—
WorkLifeBalance	Whole Number	Foreign Key
SelfRating	Whole Number	Foreign Key
ManagerRating	Whole Number	Foreign Key

In our dataset, **ManagerRating** refers to the **manager's evaluation of the employee**.

- **SelfRating** → rating the employee gives themselves.
- **ManagerRating** → rating the manager gives to the employee.

3- EducationLevel Table

1. Source Table: Education (renamed to DimEducationLevel)

Column Name	Data Type	Key Type
EducationLevelID	Whole Number	Primary Key
EducationLevel	Text	—

4- RatingLevel Table

Transformations Applied

1. **Created a reference from RatingLevel** – used to preserve the original table for multiple lookups.
2. **Renamed the reference table** – to a descriptive name for its purpose:
 - DimSelfRating → used for FactPerformanceRating.SelfRating
 - DimManagerRating → used for FactPerformanceRating.ManagerRating
3. **Validated column data types:**
 - RatingID → Whole Number (Primary Key)
 - RatingLevel → Text
4. **Resulting Tables**
 - DimSelfRating
 - DimManagerRating

DimSelfRating / DimManagerRating – Column Data Types

Column Name	Data Type	Key Type
RatingID	Whole Number	Primary Key
RatingLevel	Text	—

5- SatisfiedLevel Table

Transformations Applied

1. **Created 3 references from SatisfiedLevel** – used to preserve the original table for multiple lookups.

2. Renamed each reference table to match its purpose in FactPerformanceRating:

- DimWorkLifeBalance → linked to FactPerformanceRating.WorkLifeBalance
- DimEnvironmentSatisfaction → linked to FactPerformanceRating.EnvironmentSatisfaction
- DimJobSatisfaction → linked to FactPerformanceRating.JobSatisfaction
- DimRelationshipSatisfaction → linked to FactPerformanceRating.RelationshipSatisfaction

3. Validated column data types:

- SatisfactionID → Whole Number (Primary Key)
- SatisfactionLevel → Text

4. Resulting Tables

- DimWorkLifeBalance
- DimEnvironmentSatisfaction
- DimJobSatisfaction
- DimRelationshipSatisfaction

Column Name	Data Type	Key Type
SatisfactionID	Whole Number	Primary Key
SatisfactionLevel	Text	—

5. Power BI Data Model

1. Fact Table

FactPerformanceRating

- Contains employee performance evaluations.
- **Primary Key:** PerformanceID
- **Foreign Keys:**
 - EmployeeID → DimEmployee[EmployeeID]
 - EnvironmentSatisfaction → DimEnvironmentSatisfaction[SatisfactionID]
 - JobSatisfaction → DimJobSatisfaction[SatisfactionID]
 - RelationshipSatisfaction → DimRelationshipSatisfaction[SatisfactionID]
 - WorkLifeBalance → DimWorkLifeBalance[SatisfactionID]
 - SelfRating → DimSelfRating[RatingID]

- ManagerRating → DimManagerRating[RatingID]

2. Dimension Tables

Table Name	Primary Key	Description / Purpose
DimEmployee	EmployeeID	Contains employee demographic, job, and organizational details.
DimEducationLevel	EducationLevelID	Defines education categories or levels for employees.
DimEnvironmentSatisfaction	SatisfactionID	Maps satisfaction scores to descriptive levels for environment satisfaction.
DimJobSatisfaction	SatisfactionID	Maps satisfaction scores to descriptive levels for job satisfaction.
DimRelationshipSatisfaction	SatisfactionID	Maps satisfaction scores to descriptive levels for relationship satisfaction.
DimWorkLifeBalance	SatisfactionID	Maps satisfaction scores to descriptive levels for work–life balance.
DimSelfRating	RatingID	Describes the rating scale for self-assessment.
DimManagerRating	RatingID	Describes the rating scale for manager-assigned performance ratings.

3. Relationships Overview

From Table	From Column	To Table	To Column	Cardinality	Relationship Type
FactPerformanceRating	EmployeeID	DimEmployee	EmployeeID	Many-to-One	Regular
DimEmployee	EducationLevelID	DimEducationLevel	EducationLevelID	Many-to-One	Regular
FactPerformanceRating	EnvironmentSatisfaction	DimEnvironmentSatisfaction	SatisfactionID	Many-to-One	Regular
FactPerformanceRating	JobSatisfaction	DimJobSatisfaction	SatisfactionID	Many-to-One	Regular
FactPerformanceRating	RelationshipSatisfaction	DimRelationshipSatisfaction	SatisfactionID	Many-to-One	Regular
FactPerformanceRating	WorkLifeBalance	DimWorkLifeBalance	SatisfactionID	Many-to-One	Regular
FactPerformanceRating	SelfRating	DimSelfRating	RatingID	Many-to-One	Regular
FactPerformanceRating	ManagerRating	DimManagerRating	RatingID	Many-to-One	Regular

4. Model Summary

- **Star Schema Design** with one central fact table (FactPerformanceRating) and multiple supporting dimension tables.

- Ensures **clean separation of descriptive data** (dimensions) and **quantitative performance data** (fact).
- Optimized for performance analytics, satisfaction analysis, and employee performance dashboards.

6.Metric Group: Attrition

Purpose

This metric group measures employee retention and turnover performance across the organization. It provides insights into the number of active and inactive employees, attrition rates, and average years of service overall and by attrition status.

Defined Measures

Measure Name	DAX Formula	Description / Purpose
Active Employees	DAX Active Employees = CALCULATE([Total Employees], NOT(DimEmployee[Attrition]))	Calculates the number of employees who are currently active (i.e., have not left the organization) .
Inactive Employees	DAX Inactive Employees = CALCULATE([Total Employees], DimEmployee[Attrition])	Counts the number of employees marked as attrited (Attrition = TRUE).

Measure Name	DAX Formula	Description / Purpose
Attrition Rate	DAX Attrition Rate = DIVIDE([Inactive Employees], [Total Employees])	Computes the percentage of employees who have left the organization.
Avg YearsOfService	DAX Avg YearsOfService = AVERAGE(DimEmployee[YearsOfService])	Calculates the average number of years employees have been with the company.
AvgYearsOfService_Active	DAX AvgYearsOfService_Active = CALCULATE(AVERAGE(DimEmployee[YearsOfService]), DimEmployee[Attrition] = FALSE())	Calculates the average tenure of employees who are still active.
AvgYearsOfService_Attrited	DAX AvgYearsOfService_Attrited = CALCULATE(AVERAGE(DimEmployee[YearsOfService]), DimEmployee[Attrition] = TRUE())	Calculates the average tenure of employees who have left (attrited).

Usage Notes

- All measures are **based on the DimEmployee table** and depend on a previously defined measure [Total Employees].

- These measures are used in **Attrition Analysis visuals** such as:
 - KPI Cards (Active Employees, Inactive Employees, Attrition Rate)
 - Bar or Column Charts (Years of Service comparisons)
 - Slicers for Department, Gender, MaritalStatus, and EducationLevel to segment results.

Insights Enabled

- Compare retention across departments, genders, and age groups.
- Monitor changes in active versus inactive workforce over time.

7. Data Coverage Metrics

Purpose

The **Data Coverage Metrics** group defines the temporal range of the dataset across both employee and performance data. These measures help identify the earliest and latest recorded dates and calculate the overall data span in years.

Measures Defined

Measure Name	Formula (DAX)	Description / Purpose
Max Hire Date	<code>MAX(DimEmployee[HireDate])</code>	Latest hire date recorded in the employee dataset.
Min Review Date	<code>MIN(FactPerformanceRating[ReviewDate])</code>	Earliest performance review date recorded.
Max Review Date	<code>MAX(FactPerformanceRating[ReviewDate])</code>	Latest performance review date recorded.
Earliest Data Date	<code>MINX({ [Min Hire Date], [Min Review Date] }, [Value])</code>	Determines the earliest date across



Measure Name	Formula (DAX)	Description / Purpose
		both employee and review data.
Latest Data Date	<code>MAXX({ [Max Hire Date], [Max Review Date] }, [Value])</code>	Determines the latest date across both employee and review data.
Data Coverage (Years)	<code>DATEDIFF([Earliest Data Date], [Latest Data Date], YEAR)</code>	Calculates total span of available data (in years).

Usage in Reports

- Display **Earliest Data Date** and **Latest Data Date** in card visuals to show the active data range.
- Use **Data Coverage (Years)** in KPI cards or report headers to indicate dataset span.

8. Salary Insights Metrics

Purpose

The **Salary Insights Metrics** group provides key measures that analyze employee compensation patterns across the organization. These metrics allow HR analysts and management to compare salary levels between active and attrited employees, understand overall pay ranges, and identify possible salary-driven attrition trends.



Measures Overview

Measure Name	Formula	Description
Avg Salary	<code>AVERAGE(DimEmployee[Salary])</code>	Calculates the overall average salary across all employees in the dataset.
Avg Salary - Active	<code>CALCULATE(AVERAGE(DimEmployee[Salary]), DimEmployee[Attrition] = FALSE())</code>	Computes the average salary specifically for active employees.
Avg Salary - Attrited	<code>CALCULATE(AVERAGE(DimEmployee[Salary]), DimEmployee[Attrition] = TRUE())</code>	Computes the average salary specifically for employees who have left the organization.
Max Salary	<code>MAX(DimEmployee[Salary])</code>	Returns the highest salary recorded among all employees.
Min Salary	<code>MIN(DimEmployee[Salary])</code>	Returns the lowest salary recorded among all employees.

Usage

These measures can be applied in visuals such as:

- **KPI cards:** To show current average, max, and min salaries.

- **Bar/column charts:** To compare average salaries of active vs. attrited employees.
- **Trend charts:** When combined with time-related fields (e.g., Hire Date), to observe salary evolution over time.
- **Department or Role breakdowns:** To visualize pay distribution across organizational units.

9. Managerial Tenure Insights Metrics

Purpose

The **Managerial Tenure Insights Metrics** group focuses on analyzing the duration of the working relationship between employees and their current managers. These metrics help evaluate leadership stability, managerial influence, and potential links between manager tenure and employee outcomes such as satisfaction, performance, or attrition.

Metrics Overview

Metric Name	Formula (DAX)	Description
Min YearsWithCurrManager	MIN(DimEmployee[YearsWithCurrManager])	Identifies the shortest duration (in years) that any employee has worked with their current manager.
Avg YearsWithCurrManager	AVERAGE(DimEmployee[YearsWithCurrManager])	Calculates the average number of years employees have been under the same manager, indicating managerial stability

Metric Name	Formula (DAX)	Description
		across the organization.
Max YearsWithCurrManager	<code>MAX(DimEmployee[YearsWithCurrManager])</code>	Shows the longest duration (in years) an employee has remained with the same manager, highlighting sustained leadership relationships.

Business Insights

- Helps identify whether longer employee–manager relationships are associated with improved retention or satisfaction.
- Useful for detecting departments or teams with frequent managerial changes.
- Supports workforce planning by highlighting management continuity and leadership stability.

10. Calculated columns in DimEmployee Table

Column Name	Description / Purpose	DAX Formula
LatestReviewDate	Retrieves the most recent performance review date for each employee.	<code>MAXX(FILTER(FactPerformanceRating, FactPerformanceRating[Em</code>



Column Name	Description / Purpose	DAX Formula
		ployeeID] = DimEmployee[EmployeeID]), FactPerformanceRating[Revi ewDate])
DaysBetweenHireAndLat estReview	Calculates the number of days between an employee's hire date and their most recent review.	IF(ISBLANK([LatestReviewD ate]), BLANK(), DATEDIFF([HireDate], [LatestReviewDate], DAY))
YearsOfService	Total years of service for each employee based on hire date and latest review.	DATEDIFF([HireDate], [LatestReviewDate], YEAR)
YearsAtCompanyConsist ency	Validates whether recorded YearsAtCompan y matches computed YearsOfServic e.	IF([YearsAtCompany] = [YearsOfService], "Match", "Mismatch")
AgeGroup	Groups employees into standardized age brackets.	SWITCH(TRUE(), [Age] < 25, "Under 25", [Age] < 35, "25-34", [Age] < 45, "35-44", [Age] < 55, "45-54", "55+")
TenureBand	Categorizes employees based on total years of service.	SWITCH(TRUE(), [YearsOfService] < 1, "<1", [YearsOfService] < 3, "1-2", [YearsOfService] < 5, "3-4", [YearsOfService] < 10, "5-9", "10+")
IsRecentlyPromoted	Flags employees promoted in the last 2 years.	IF([YearsSinceLastPromotio n] <= 2, TRUE(), FALSE())



Column Name	Description / Purpose	DAX Formula
HasLongRelationshipWithManager	Flags employees with current manager ≥ 3 years.	<code>IF([YearsWithCurrManager] >= 3, TRUE(), FALSE())</code>
CheckDiff	Difference between calculated YearsOfService and stored YearsAtCompany.	<code>[YearsOfService] - [YearsAtCompany]</code>
EstimatedLastDate	Estimates the last day of the employee's most recent completed work year.	<code>DATE(YEAR([HireDate]) + [YearsAtCompany] + 1, MONTH([HireDate]), DAY([HireDate])) - 1</code>
ManagerTenureGroup	Groups employees by tenure with current manager.	<code>SWITCH(TRUE(), [YearsWithCurrManager] = 0, "0 Years", [YearsWithCurrManager] <= 2, "1-2 Years", [YearsWithCurrManager] <= 4, "3-4 Years", [YearsWithCurrManager] <= 6, "5-6 Years", [YearsWithCurrManager] <= 8, "7-8 Years", [YearsWithCurrManager] <= 10, "9-10 Years", "10+ Years")</code>
EmployeeYearsAtCompanyGroup	Groups employees by total years at the company.	<code>SWITCH(TRUE(), [YearsAtCompany] = 0, "0 Years", [YearsAtCompany] <= 2, "1-2 Years", [YearsAtCompany] <= 4, "3-4 Years", [YearsAtCompany] <= 6, "5-6 Years", [YearsAtCompany] <= 8, "7-8 Years", [YearsAtCompany] > 8, "9+ Years")</code>



Column Name	Description / Purpose	DAX Formula
		<code><= 10, "9-10 Years", "10+ Years")</code>
PromotionGapGroup	Groups employees by time since last promotion.	<code>SWITCH(TRUE(), [YearsSinceLastPromotion] = 0, "0 Years", [YearsSinceLastPromotion] ≤ 2, "1-2 Years", [YearsSinceLastPromotion] ≤ 4, "3-4 Years", [YearsSinceLastPromotion] ≤ 6, "5-6 Years", "6+ Years")</code>
DistanceCategory	Groups employees based on distance from home to workplace.	<code>SWITCH(TRUE(), [DistanceFromHome (KM)] ≤ 5, "0-5 KM", [DistanceFromHome (KM)] ≤ 15, "6-15 KM", [DistanceFromHome (KM)] ≤ 25, "16-25 KM", [DistanceFromHome (KM)] ≤ 35, "26-35 KM", "35+ KM")</code>
SalaryGroup	Groups employees by salary range.	<code>SWITCH(TRUE(), [Salary] < 40000, "Very Low", [Salary] ≤ 70000, "Low", [Salary] ≤ 120000, "Medium", [Salary] ≤ 250000, "High", "Very High")</code>
YearsInRoleGroup	Groups employees by years in most recent role.	<code>SWITCH(TRUE(), [YearsInMostRecentRole] = 0, "0 Years", [YearsInMostRecentRole] ≤ 1, "1 Year", [YearsInMostRecentRole] ≤ 3, "2-3 Years",</code>



Column Name	Description / Purpose	DAX Formula
		[YearsInMostRecentRole] <= 5, "4-5 Years", "5+ Years")

A. LatestReviewDate

Formula: LatestReviewDate = MAXX(FILTER(FactPerformanceRating, FactPerformanceRating[EmployeeID] = DimEmployee[EmployeeID]), FactPerformanceRating[ReviewDate])

Purpose: Retrieves the most recent performance review date for each employee from the FactPerformanceRating table.

Use Case: Used to calculate service duration, consistency checks, and identify employees with outdated reviews.

B. DaysBetweenHireAndLatestReview

Formula: DaysBetweenHireAndLatestReview = IF(ISBLANK([LatestReviewDate]), BLANK(), DATEDIFF([HireDate], [LatestReviewDate], DAY))

Purpose: Calculates the number of days between an employee's hire date and their most recent review date.

Use Case: Useful for understanding employee tenure duration at the time of their last evaluation.

C. YearsOfService

Formula: YearsOfService = DATEDIFF([HireDate], [LatestReviewDate], YEAR)

Purpose: Calculates total years of service for each employee based on the hire date and their latest review.

Use Case: Used in tenure-based grouping and comparison metrics (e.g., employee experience, retention analysis).

D. YearsAtCompanyConsistency

Formula: YearsAtCompanyConsistency = IF([YearsAtCompany] = [YearsOfService], "Match", "Mismatch")

Purpose: Validates whether the recorded YearsAtCompany aligns with the computed YearsOfService.

Use Case: Data quality check to ensure consistency between derived and stored tenure values.

E. AgeGroup

Formula: AgeGroup = SWITCH(TRUE(), [Age] < 25, "Under 25", [Age] < 35, "25-34", [Age] < 45, "35-44", [Age] < 55, "45-54", "55+")

Purpose: Groups employees into standardized age brackets.

Use Case: Demographic analysis and age-based performance or attrition comparisons.

F. TenureBand

Formula: TenureBand = SWITCH(TRUE(), [YearsOfService] < 1, "<1", [YearsOfService] < 3, "1-2", [YearsOfService] < 5, "3-4", [YearsOfService] < 10, "5-9", "10+")

Purpose: Categorizes employees based on total years of service.

Use Case: Used for visual grouping in dashboards or tenure-based insights.

G. IsRecentlyPromoted

Formula: IsRecentlyPromoted = IF([YearsSinceLastPromotion] <= 2, TRUE(), FALSE())

Purpose: Flags employees who have been promoted in the last two years.

Use Case: Promotion tracking and career progression analysis.

H. HasLongRelationshipWithManager

Formula: HasLongRelationshipWithManager = IF([YearsWithCurrManager] >= 3, TRUE(), FALSE())

Purpose: Indicates whether an employee has been under the same manager for three or more years.

Use Case: Used to study employee-manager relationship impact on performance and retention.

I. CheckDiff

Formula: CheckDiff = [YearsOfService] - [YearsAtCompany]

Purpose: Computes the difference between derived and reported years at the company.

Use Case: Data integrity validation for tenure-related metrics.

J. EstimatedLastDate

Formula: EstimatedLastDate = DATE(YEAR(DimEmployee[HireDate]) + DimEmployee[YearsAtCompany] + 1, MONTH(DimEmployee[HireDate]), DAY(DimEmployee[HireDate])) - 1



Purpose: Estimates the last day of an employee's most recent completed work year.

Use Case: Used for retrospective analysis or approximating service milestones.

K. ManagerTenureGroup

Formula: ManagerTenureGroup = SWITCH(TRUE(),
DimEmployee[YearsWithCurrManager] = 0, "0 Years",
DimEmployee[YearsWithCurrManager] <= 2, "1-2 Years",
DimEmployee[YearsWithCurrManager] <= 4, "3-4 Years",
DimEmployee[YearsWithCurrManager] <= 6, "5-6 Years",
DimEmployee[YearsWithCurrManager] <= 8, "7-8 Years",
DimEmployee[YearsWithCurrManager] <= 10, "9-10 Years", "10+ Years")

Purpose: Groups employees based on the number of years they've been with their current manager.

Use Case: Managerial relationship tenure analysis and stability assessment.

L. EmployeeYearsAtCompanyGroup

Formula: EmployeeYearsAtCompanyGroup = SWITCH(TRUE(),
DimEmployee[YearsAtCompany] = 0, "0 Years", DimEmployee[YearsAtCompany] <= 2,
"1-2 Years", DimEmployee[YearsAtCompany] <= 4, "3-4 Years",
DimEmployee[YearsAtCompany] <= 6, "5-6 Years", DimEmployee[YearsAtCompany]
<= 8, "7-8 Years", DimEmployee[YearsAtCompany] <= 10, "9-10 Years", "10+ Years")

Purpose: Categorizes employees based on total years at the company.

Use Case: Experience segmentation for retention, satisfaction, or performance dashboards.

M. PromotionGapGroup

Formula: PromotionGapGroup = SWITCH(TRUE(),
DimEmployee[YearsSinceLastPromotion] = 0, "0 Years",
DimEmployee[YearsSinceLastPromotion] <= 2, "1-2 Years",
DimEmployee[YearsSinceLastPromotion] <= 4, "3-4 Years",
DimEmployee[YearsSinceLastPromotion] <= 6, "5-6 Years", "6+ Years")

Purpose: Groups employees by time since their last promotion.

Use Case: Promotion frequency analysis and career growth trend evaluation.

N. DistanceCategory

Formula: DistanceCategory = SWITCH(TRUE(), DimEmployee[DistanceFromHome (KM)] <= 5, "0-5 KM", DimEmployee[DistanceFromHome (KM)] <= 15, "6-15 KM",
DimEmployee[DistanceFromHome (KM)] <= 25, "16-25 KM",
DimEmployee[DistanceFromHome (KM)] <= 35, "26-35 KM", "35+ KM")

Purpose: Classifies employees based on their distance from home to workplace.

Use Case: Used in commute impact studies or geographic employee segmentation.

O. SalaryGroup

Formula: SalaryGroup = SWITCH(TRUE(), DimEmployee[Salary] < 40000, "Very Low", DimEmployee[Salary] <= 70000, "Low", DimEmployee[Salary] <= 120000, "Medium", DimEmployee[Salary] <= 250000, "High", "Very High")

Purpose: Groups employees by salary range.

Use Case: Income distribution analysis, compensation fairness, and pay grade comparisons.

P. YearsInRoleGroup

Formula: YearsInRoleGroup = SWITCH(TRUE(), DimEmployee[YearsInMostRecentRole] = 0, "0 Years", DimEmployee[YearsInMostRecentRole] <= 1, "1 Year", DimEmployee[YearsInMostRecentRole] <= 3, "2-3 Years", DimEmployee[YearsInMostRecentRole] <= 5, "4-5 Years", "5+ Years")

Purpose: Groups employees by the duration spent in their current role.

Use Case: Role tenure analysis and potential role stagnation tracking.

11. FactPerformanceRating – Calculated Columns Documentation

Column Name	DAX Formula / Logic	Description / Purpose
ValidReview	IF(FactPerformanceRating[ReviewDate] >= RELATED(DimEmployee[HireDate]) && FactPerformanceRating[ReviewDate] <= RELATED(DimEmployee[EstimatedLastDate]), TRUE(), FALSE())	Flags whether the review date falls within the employee's employment period.
OverallSatisfactionScore	([EnvironmentSatisfaction] + [JobSatisfaction] + [RelationshipSatisfaction] + [WorkLifeBalance]) / 4	Calculates the average satisfaction score across four dimensions.



Column Name	DAX Formula / Logic	Description / Purpose
OverallRatingScore	$([SelfRating] + [ManagerRating]) / 2$	Calculates the average overall rating combining self-assessment and manager rating.
IsHighSatisfaction	$IF([OverallSatisfactionScore] \geq 4, TRUE(), FALSE())$	Flags reviews where overall satisfaction is high .
IsLowSatisfaction	$IF([OverallSatisfactionScore] \leq 2, TRUE(), FALSE())$	Flags reviews where overall satisfaction is low .
IsHighRating	$IF([OverallRatingScore] \geq 4, TRUE(), FALSE())$	Flags reviews where overall rating is high .
IsLowRating	$IF([OverallRatingScore] \leq 2, TRUE(), FALSE())$	Flags reviews where overall rating is low .

12. Performance Flags Metrics (FactPerformanceRating)

Metric Name	Description / Purpose	DAX Formula
High_EnvironmentSatisfaction_Reviews	Counts reviews where employee reported	$COUNTROWS(FILTER(FactPerformanceRating, FactPerformanceRating[EnvironmentSatisfaction] \geq 4))$

Metric Name	Description / Purpose	DAX Formula
	high environment satisfaction (≥ 4).	
Low_EnvironmentSatisfaction_Reviews	Counts reviews where employee reported low environment satisfaction (≤ 2).	<code>COUNTROWS(FILTER(FactPerformanceRating, FactPerformanceRating[EnvironmentSatisfaction] <= 2))</code>
High_JobSatisfaction_Reviews	Counts reviews where employee reported high job satisfaction (≥ 4).	<code>COUNTROWS(FILTER(FactPerformanceRating, FactPerformanceRating[JobSatisfaction] >= 4))</code>
Low_JobSatisfaction_Reviews	Counts reviews where employee reported low job satisfaction (≤ 2).	<code>COUNTROWS(FILTER(FactPerformanceRating, FactPerformanceRating[JobSatisfaction] <= 2))</code>
High_RelationshipSatisfaction_Reviews	Counts reviews where	<code>COUNTROWS(FILTER(FactPerformanceRating,</code>

Metric Name	Description / Purpose	DAX Formula
	employee reported high relationship satisfaction (≥ 4).	FactPerformanceRating[RelationshipSatisfaction] ≥ 4))
Low_RelationshipSatisfaction_Reviews	Counts reviews where employee reported low relationship satisfaction (≤ 2).	COUNTROWS(FILTER(FactPerformanceRating, FactPerformanceRating[RelationshipSatisfaction] ≤ 2))
High_WorkLifeBalance_Reviews	Counts reviews where employee reported high work-life balance (≥ 4).	COUNTROWS(FILTER(FactPerformanceRating, FactPerformanceRating[WorkLifeBalance] ≥ 4))
Low_WorkLifeBalance_Reviews	Counts reviews where employee reported low work-life balance (≤ 2).	COUNTROWS(FILTER(FactPerformanceRating, FactPerformanceRating[WorkLifeBalance] ≤ 2))



Metric Name	Description / Purpose	DAX Formula
High_SelfRating_Reviews	Counts reviews where employee's self-rating is high (≥ 4).	<code>COUNTROWS(FILTER(FactPerformanceRating, FactPerformanceRating[SelfRating] >= 4))</code>
Low_SelfRating_Reviews	Counts reviews where employee's self-rating is low (≤ 2).	<code>COUNTROWS(FILTER(FactPerformanceRating, FactPerformanceRating[SelfRating] <= 2))</code>
High_ManagerRating_Reviews	Counts reviews where manager gave high rating (≥ 4).	<code>COUNTROWS(FILTER(FactPerformanceRating, FactPerformanceRating[ManagerRating] >= 4))</code>
Low_ManagerRating_Reviews	Counts reviews where manager gave low rating (≤ 2).	<code>COUNTROWS(FILTER(FactPerformanceRating, FactPerformanceRating[ManagerRating] <= 2))</code>

Purpose / Use Case:



- Provides quick insight into high and low satisfaction or performance ratings per dimension.
- Can be used in dashboards to identify strengths and areas needing attention across employees and departments.
- Useful for HR analysis, employee engagement monitoring, and performance review trends.

13. Project results and outcome

This project produced a complete HR analytics solution using Power BI. The raw data was cleaned, organized, and modeled into a clear structure, allowing key HR metrics to be visualized effectively. The final dashboard provides insights into employee attrition, satisfaction, performance, salary patterns, and promotion trends.

With these visual tools, HR managers can now identify problem areas quickly, monitor workforce trends, and make informed decisions based on accurate, real-time data. The outcome is a practical and accessible system that supports better planning, improves employee retention, and enhances overall HR strategy.

14. Proposed solutions

Reduce Attrition

- Identify high-risk groups using attrition metrics.
- Improve work-life balance through flexible hours or workload adjustments.
- Strengthen manager–employee relationships with better communication and leadership training.

Improve Employee Satisfaction

- Monitor departments with low satisfaction scores.
- Provide regular feedback sessions between employees and managers.
- Offer wellness programs and promote balanced workloads.

Enhance Compensation Fairness

- Review salary differences across roles, gender, and departments.
- Align pay with experience, tenure, and performance.
- Improve overtime policies and compensation.

Support Career Growth

- Expand training programs and link them to promotions.
- Track employees stuck in the same role for long periods.

- Provide clear career paths and upskilling opportunities.

Strengthen Diversity & Inclusion

- Monitor workforce diversity metrics.
- Address performance or pay gaps among demographic groups.
- Standardize evaluation criteria to reduce bias.

Improve HR Decision-Making

- Use Power BI dashboards regularly for real-time insights.
- Maintain data quality through periodic data checks.
- Centralize HR reports to support strategic planning.

15. Conclusion

This project showed how HR data can be turned into useful insights with Power BI. By cleaning, modeling, and analyzing the data, we were able to understand important factors that affect employee attrition, satisfaction, performance, and salary fairness.

The dashboard makes it easy for HR teams to see trends, identify issues, and make better decisions. Overall, the project provides a practical tool that can help the organization improve employee retention, support growth, and create a more positive and efficient work environment.