-- Create a new database named HR\_Analysis\_DataBase

CREATE DATABASE HR\_Analysis\_DataBase

-- Use the newly created database

USE HR\_Analysis\_DataBase

----------------------------- Week 1: Build Data Model, Data Cleaning and Preprocessing ------------------

----------------------------------------------------------------------------------------------------------

-- Step 1: Import CSV Files

-- import Employee table

-- Righ click on HR\_Analysis\_DataBase database >>>> import wizard

-- Convert Data Type (OverTime Column) into nvarchar(50)

-- Convert Data Type (Attrition) into nvarchar(50)

select \* from Employee

-- import PerformanceRating table

-- Righ click on HR\_Analysis\_DataBase database >>>> import wizard

-- Convert Data Type (ReviewData Column) into nvarchar(50), because an error occured !!

select \* from PerformanceRating

-- Re-Converte Data Type (ReviewData Column) into date !!

SELECT COLUMN\_NAME, DATA\_TYPE

FROM INFORMATION\_SCHEMA.COLUMNS

WHERE TABLE\_NAME = 'PerformanceRating'

ALTER TABLE PerformanceRating

ALTER COLUMN ReviewDate DATE;

-- import EducationLevel table

-- Righ click on HR\_Analysis\_DataBase database >>>> import wizard

select \* from EducationLevel

-- import RatingLevel table

-- Righ click on HR\_Analysis\_DataBase database >>>> import wizard

select \* from RatingLevel

-- import SatisfiedLevel table

-- Righ click on HR\_Analysis\_DataBase database >>>> import wizard

select \* from SatisfiedLevel

----------------------------- Relationships Setup -----------------------------------

-- Ensure the tables have proper primary keys and relationships if available

-- Adding sample relationships based on assumed structure

-- Assuming EmployeeID as Primary Key in Employee table

ALTER TABLE Employee

ADD CONSTRAINT PK\_Employee PRIMARY KEY (EmployeeID);

-- Assuming PerformanceRatingID as Primary Key in PerformanceRating table

ALTER TABLE PerformanceRating

ADD CONSTRAINT PK\_PerformanceRating PRIMARY KEY (PerformanceID, EmployeeID);

-- Assuming EducatioID as Primary Key in EducationLevel table

ALTER TABLE EducationLevel

ADD CONSTRAINT PK\_EducationLevel PRIMARY KEY (EducationLevelID);

-- Assuming RatingLevelID as Primary Key in RatingLevel table

ALTER TABLE RatingLevel

ADD CONSTRAINT PK\_RatingLevelID PRIMARY KEY (RatingID);

-- Assuming SatisfactionID as Primary Key in SatisfiedLevel table

ALTER TABLE SatisfiedLevel

ADD CONSTRAINT PK\_SatisfiedLevel PRIMARY KEY (SatisfactionID);

-- Establishing Relationships

-- Merging Primary Tables

-- Merge Employee table and PerformanceRating table using EmployeeID

SELECT e.\*, p.\*

FROM Employee e

LEFT JOIN PerformanceRating p ON e.EmployeeID = p.EmployeeID;

-- Merging Secondary Tables

-- Merge with Education Level (mapping EducationLevelID to Education)

SELECT e.\*, p.\*, el.EducationLevel

FROM Employee e

LEFT JOIN PerformanceRating p ON e.EmployeeID = p.EmployeeID

LEFT JOIN EducationLevel el ON e.Education = el.EducationLevelID;

-- Merge with Satisfaction Level (mapping EnvironmentSatisfaction to SatisfactionID)

SELECT e.\*, p.\*, el.EducationLevel, sl.SatisfactionLevel

FROM Employee e

LEFT JOIN PerformanceRating p ON e.EmployeeID = p.EmployeeID

LEFT JOIN EducationLevel el ON e.Education = el.EducationLevelID

LEFT JOIN SatisfiedLevel sl ON p.EnvironmentSatisfaction = sl.SatisfactionID;

-- Merge with Rating Level (mapping ManagerRating to RatingLevelID)

SELECT e.\*, p.\*, el.EducationLevel, sl.SatisfactionLevel, rl.RatingLevel

FROM Employee e

LEFT JOIN PerformanceRating p ON e.EmployeeID = p.EmployeeID

LEFT JOIN EducationLevel el ON e.Education = el.EducationLevelID

LEFT JOIN SatisfiedLevel sl ON p.EnvironmentSatisfaction = sl.SatisfactionID

LEFT JOIN RatingLevel rl ON p.ManagerRating = rl.RatingID;

--------------------- Ending of Week 1: Build Data Model, Data Cleaning and Preprocessing

----------------------------- Week 2: Analysis Questions Phase -----------------------------------

--------------------------------------------------------------------------------------------------

-- First Categoty

-- Employee Demographics & Salary Analysis

-- 1. What is the distribution of employees by education level, job role, and department?

SELECT

el.EducationLevel, e.JobRole, e.Department,

COUNT(\*) AS EmployeeCount

FROM Employee e

LEFT JOIN EducationLevel el ON e.Education = el.EducationLevelID

GROUP BY el.EducationLevel, e.JobRole, e.Department;

-- 2. How does the average salary vary by education level?

SELECT el.EducationLevel,

AVG(e.Salary) AS AverageSalary

FROM Employee e

LEFT JOIN EducationLevel el ON e.Education = el.EducationLevelID

GROUP BY el.EducationLevel;

-- 3. Is there a gender pay gap across different job roles and departments?

SELECT e.JobRole, e.Department, e.Gender,

AVG(e.Salary) AS AverageSalary

FROM Employee e

GROUP BY e.JobRole, e.Department, e.Gender;

-- 4. What is the salary distribution based on years of experience?

SELECT e.YearsAtCompany,

AVG(e.Salary) AS AverageSalary,

COUNT(\*) AS EmployeeCount

FROM Employee e

GROUP BY e.YearsAtCompany;

-- 5. Which departments have the highest and lowest average salaries?

SELECT e.Department,

AVG(e.Salary) AS AverageSalary

FROM Employee e

GROUP BY e.Department

ORDER BY AverageSalary DESC;

-- Second Categoty

-- Employee Satisfaction & Engagement

-- 6. What is the average satisfaction level across different job roles?

SELECT e.JobRole,

AVG(p.JobSatisfaction) AS AverageSatisfaction,

CASE

WHEN AVG(p.JobSatisfaction) = 1 THEN 'Very Dissatisfied'

WHEN AVG(p.JobSatisfaction) = 2 THEN 'Dissatisfied'

WHEN AVG(p.JobSatisfaction) = 3 THEN 'Neutral'

WHEN AVG(p.JobSatisfaction) = 4 THEN 'Satisfied'

WHEN AVG(p.JobSatisfaction) = 5 THEN 'Very Satisfied'

ELSE 'Unknown'

END AS SatisfactionLevel

FROM Employee e

LEFT JOIN PerformanceRating p ON e.EmployeeID = p.EmployeeID

LEFT JOIN SatisfiedLevel sl ON p.JobSatisfaction = sl.SatisfactionID

GROUP BY e.JobRole;

-- 7. Is there a relationship between satisfaction level and salary?

SELECT

AVG(e.Salary) AS AverageSalary,

AVG(CASE

WHEN sl.SatisfactionLevel = 'Very Dissatisfied' THEN 1

WHEN sl.SatisfactionLevel = 'Dissatisfied' THEN 2

WHEN sl.SatisfactionLevel = 'Neutral' THEN 3

WHEN sl.SatisfactionLevel = 'Satisfied' THEN 4

WHEN sl.SatisfactionLevel = 'Very Satisfied' THEN 5

ELSE NULL

END) AS AverageSatisfaction

FROM Employee e

LEFT JOIN PerformanceRating p ON e.EmployeeID = p.EmployeeID

LEFT JOIN SatisfiedLevel sl ON p.JobSatisfaction = sl.SatisfactionID;

-- 8. Do employees with higher education levels report higher satisfaction?

SELECT el.EducationLevel,

AVG(CASE

WHEN sl.SatisfactionLevel = 'Very Dissatisfied' THEN 1

WHEN sl.SatisfactionLevel = 'Dissatisfied' THEN 2

WHEN sl.SatisfactionLevel = 'Neutral' THEN 3

WHEN sl.SatisfactionLevel = 'Satisfied' THEN 4

WHEN sl.SatisfactionLevel = 'Very Satisfied' THEN 5

ELSE NULL

END) AS AverageSatisfaction

FROM Employee e

LEFT JOIN EducationLevel el ON e.Education = el.EducationLevelID

LEFT JOIN PerformanceRating p ON e.EmployeeID = p.EmployeeID

LEFT JOIN SatisfiedLevel sl ON p.JobSatisfaction = sl.SatisfactionID

GROUP BY el.EducationLevel;

-- 9. Which departments have the most satisfied and least satisfied employees?

SELECT e.Department,

AVG(CASE

WHEN sl.SatisfactionLevel = 'Very Dissatisfied' THEN 1

WHEN sl.SatisfactionLevel = 'Dissatisfied' THEN 2

WHEN sl.SatisfactionLevel = 'Neutral' THEN 3

WHEN sl.SatisfactionLevel = 'Satisfied' THEN 4

WHEN sl.SatisfactionLevel = 'Very Satisfied' THEN 5

ELSE NULL

END) AS AverageSatisfaction

FROM Employee e

LEFT JOIN PerformanceRating p ON e.EmployeeID = p.EmployeeID

LEFT JOIN SatisfiedLevel sl ON p.JobSatisfaction = sl.SatisfactionID

GROUP BY e.Department

ORDER BY AverageSatisfaction DESC;

-- 10. Does job role impact satisfaction level?

SELECT e.JobRole,

AVG(CASE

WHEN sl.SatisfactionLevel = 'Very Dissatisfied' THEN 1

WHEN sl.SatisfactionLevel = 'Dissatisfied' THEN 2

WHEN sl.SatisfactionLevel = 'Neutral' THEN 3

WHEN sl.SatisfactionLevel = 'Satisfied' THEN 4

WHEN sl.SatisfactionLevel = 'Very Satisfied' THEN 5

ELSE NULL

END) AS AverageSatisfaction

FROM Employee e

LEFT JOIN PerformanceRating p ON e.EmployeeID = p.EmployeeID

LEFT JOIN SatisfiedLevel sl ON p.JobSatisfaction = sl.SatisfactionID

GROUP BY e.JobRole;

-- Third Categoty

-- Attrition & Turnover Analysis

-- 11. What is the overall employee attrition rate?

SELECT

COUNT(\*) AS TotalEmployees,

SUM(CASE WHEN e.Attrition = 'Yes' THEN 1 ELSE 0 END) AS AttritionEmployees,

(SUM(CASE WHEN e.Attrition = 'Yes' THEN 1 ELSE 0 END) \* 100.0) / COUNT(\*) AS AttritionRate

FROM Employee e;

-- 12. Which department has the highest employee turnover?

SELECT TOP 1

e.Department,

COUNT(\*) AS TotalEmployees,

SUM(CASE WHEN e.Attrition = 'Yes' THEN 1 ELSE 0 END) AS AttritionEmployees,

(SUM(CASE WHEN e.Attrition = 'Yes' THEN 1 ELSE 0 END) \* 100.0) / COUNT(\*) AS AttritionRate

FROM Employee e

GROUP BY e.Department

ORDER BY AttritionRate DESC;

-- 13. Is there a connection between satisfaction level and attrition?

SELECT sl.SatisfactionLevel,

COUNT(\*) AS TotalEmployees,

SUM(CASE WHEN e.Attrition = 'Yes' THEN 1 ELSE 0 END) AS AttritionEmployees,

(SUM(CASE WHEN e.Attrition = 'Yes' THEN 1 ELSE 0 END) \* 100.0) / COUNT(\*) AS AttritionRate

FROM PerformanceRating p

LEFT JOIN SatisfiedLevel sl ON p.EnvironmentSatisfaction = sl.SatisfactionID

LEFT JOIN Employee e ON p.EmployeeID = e.EmployeeID

GROUP BY sl.SatisfactionLevel

ORDER BY AttritionRate DESC;

-- 14. Do employees with higher education levels have lower attrition rates?

SELECT el.EducationLevel,

COUNT(\*) AS TotalEmployees,

SUM(CASE WHEN e.Attrition = 'Yes' THEN 1 ELSE 0 END) AS AttritionEmployees,

(SUM(CASE WHEN e.Attrition = 'Yes' THEN 1 ELSE 0 END) \* 100.0) / COUNT(\*) AS AttritionRate

FROM Employee e

LEFT JOIN EducationLevel el ON e.Education = el.EducationLevelID

GROUP BY el.EducationLevel

ORDER BY AttritionRate ASC;

-- 15. How does tenure (years at company) impact attrition?

SELECT

CASE

WHEN DATEDIFF(YEAR, e.HireDate, GETDATE()) BETWEEN 0 AND 1 THEN '0-1 Year'

WHEN DATEDIFF(YEAR, e.HireDate, GETDATE()) BETWEEN 2 AND 3 THEN '2-3 Years'

WHEN DATEDIFF(YEAR, e.HireDate, GETDATE()) BETWEEN 4 AND 5 THEN '4-5 Years'

ELSE '5+ Years'

END AS TenureRange,

COUNT(\*) AS TotalEmployees,

SUM(CASE WHEN e.Attrition = 'Yes' THEN 1 ELSE 0 END) AS AttritionEmployees,

(SUM(CASE WHEN e.Attrition = 'Yes' THEN 1 ELSE 0 END) \* 100.0) / COUNT(\*) AS AttritionRate

FROM Employee e

GROUP BY

CASE

WHEN DATEDIFF(YEAR, e.HireDate, GETDATE()) BETWEEN 0 AND 1 THEN '0-1 Year'

WHEN DATEDIFF(YEAR, e.HireDate, GETDATE()) BETWEEN 2 AND 3 THEN '2-3 Years'

WHEN DATEDIFF(YEAR, e.HireDate, GETDATE()) BETWEEN 4 AND 5 THEN '4-5 Years'

ELSE '5+ Years'

END

ORDER BY TenureRange;

-- Fourth Category

-- Promotion & Career Growth

-- 16. How long does it take, on average, for employees to receive a promotion?

SELECT

AVG(e.YearsSinceLastPromotion) AS AveragePromotionTime

FROM Employee e;

-- 17. Is there a correlation between education level and promotion frequency?

SELECT el.EducationLevel,

COUNT(CASE WHEN e.YearsSinceLastPromotion < 1 THEN 1 END) AS PromotionFrequency

FROM Employee e

LEFT JOIN EducationLevel el ON e.Education = el.EducationLevelID

GROUP BY el.EducationLevel

ORDER BY PromotionFrequency DESC;

-- 18. Which departments promote employees the fastest and the slowest?

SELECT e.Department,

AVG(e.YearsSinceLastPromotion) AS AveragePromotionTime

FROM Employee e

GROUP BY e.Department

ORDER BY AveragePromotionTime ASC; -- Fastest promotion first

-- 19. What percentage of satisfied employees receive promotions?

-- 19. What percentage of satisfied employees receive promotions?

WITH MedianSatisfaction AS (

-- Calculate the Median of JobSatisfaction using PERCENTILE\_CONT

SELECT

PERCENTILE\_CONT(0.5) WITHIN GROUP (ORDER BY JobSatisfaction) OVER () AS MedianJobSatisfaction

FROM PerformanceRating

)

SELECT

-- Calculate the percentage of satisfied employees who received promotions

(COUNT(CASE WHEN pr.JobSatisfaction > ms.MedianJobSatisfaction AND e.YearsSinceLastPromotion = 0 THEN 1 END) \* 100.0) /

COUNT(CASE WHEN pr.JobSatisfaction > ms.MedianJobSatisfaction THEN 1 END) AS PromotionPercentage

FROM Employee e

LEFT JOIN PerformanceRating pr ON e.EmployeeID = pr.EmployeeID

CROSS JOIN MedianSatisfaction ms;

-- 20. Does gender impact promotion opportunities?

SELECT

e.Gender,

COUNT(CASE WHEN e.YearsSinceLastPromotion = 0 THEN 1 END) AS PromotionFrequency

FROM Employee e

GROUP BY e.Gender

ORDER BY PromotionFrequency DESC;