Databricks Use Case: Employee Salary and Demographic Insights

# Introduction

In today’s data-driven HR landscape, organizations are increasingly relying on data analytics to make informed decisions related to employee engagement, compensation planning, and workforce optimization. This use case utilizes Apache Spark on Databricks to analyze employee demographic and salary data using basic PySpark DataFrame operations. The analysis aims to help HR professionals and department heads understand salary trends, age distributions, and department-level workforce metrics in an organization.

# Functional Objective

The primary objective is to analyze employee data for the following:  
- Identify high-performing cities and departments in terms of average salary.  
- Segment employees by age group to understand workforce demographics.  
- Join and enrich employee data with department information for comprehensive reporting.  
- Extract insights using SQL and PySpark DataFrame operations.

# Business Use Cases

|  |  |
| --- | --- |
| Business Scenario | Description |
| Salary Band Classification | Identify employees earning above ₹60,000 for bonus eligibility |
| Department-wise Workforce Insights | Analyze headcount and average salary per department |
| Age-Based Segmentation | Segment workforce into young (<30), mid-level (30–40), and senior (>40) |
| City-Level Performance Metrics | Find cities with the highest concentration of high earners |
| Integration with HR Reporting Tools | Export refined datasets for Power BI or HR dashboards |

# Expected Results

- A report listing average salary and employee count for each department and city.  
- A filtered dataset of high-earning employees (salary > ₹60,000).  
- A temporary view allowing SQL-based queries for ad-hoc exploration.  
- A dataset enriched with department names for business-facing reporting.  
- Clear age group classification aiding in age diversity analysis.

# Conclusion

This beginner-level Databricks exercise provides a solid foundation for real-world HR data analysis using PySpark. With simple DataFrame commands such as filter, groupBy, join, and select, users can derive actionable insights from employee data. These insights can support HR decision-making, improve compensation strategies, and assist with workforce planning. It also prepares learners to explore more advanced use cases in big data analytics using Databricks.

# Sample Data

emp.csv

id,name,age,city,salary,dept\_id  
1,Alice,28,Delhi,50000,101  
2,Bob,32,Mumbai,62000,102  
3,Charlie,24,Bangalore,45000,102  
4,David,45,Delhi,80000,101  
5,Eva,38,Mumbai,72000,103  
6,Frank,30,Chennai,56000,103

departments.csv

dept\_id,dept\_name  
101,HR  
102,Engineering  
103,Marketing