# PySpark DataFrame Hands-On Lab (Beginner Level)

---

## Datasets (Create CSV Files)

employees.csv
emp_id,name,age,salary,dept_id,joining_date
101,Ravi,25,30000,1,2022-01-10
102,Anu,28,35000,2,2021-03-15
103,Kumar,30,40000,1,2020-07-20
104,Neha,26,32000,3,2022-09-05
105,Arjun,29,28000,2,2023-02-11
106,Meena,31,45000,1,2019-12-01
107,Sneha,27,37000,4,2021-11-23
108,John,32,50000,,2020-05-18

departments.csv
dept_id,dept_name,location
1,IT,Bangalore
2,HR,Hyderabad
3,Sales,Mumbai
5,Finance,Delhi

salaries_history.csv
emp_id,year,increment
101,2023,2000
101,2024,3000
102,2023,1500
103,2023,2500
105,2024,1800
109,2023,2200

## Hands-On Questions (No Solutions)

1. Load all CSV files with header and inferSchema enabled.

2. Display schema of employees table.

3. Rename column dept_id to department_id.

4. Add column bonus (12% of salary) using withColumn().

5. Add column salary_after_bonus.

6. Drop column age.

7. Convert employee name to uppercase.

8. Extract first 3 letters of name.

9. Combine name and department_id into one column.

10. Replace department_id value 4 with null.

11. Create salary_grade column using conditions.

12. Create is_high_paid column (salary > 35000).

13. Convert joining_date to date type.

14. Extract joining year.

15. Calculate years of service till today.

16. Find employees joined after 2021.

17. Replace null department_id with 0.

18. Find employees with null department_id.

19. Drop rows where department_id is null.

20. Find total salary.

21. Find average salary.

22. Find maximum salary department-wise.

23. Count employees department-wise.

24. Find departments where average salary > 35000.

25. Perform inner join between employees and departments.

26. Perform left join and identify employees without department.

27. Find departments with no employees.

28. Join employees and salaries_history.

29. Find total increment per employee.

30. Find total increment paid in 2023.

31. Add salary category column (Very High/High/Normal).

32. Create salary_in_lakhs column.

33. Rename multiple columns appropriately.

34. Sort employees by joining_date descending.

35. Find top 3 highest paid employees.

## Classroom Instructions:

- Use only DataFrame API (No Spark SQL).

- Use withColumn() and withColumnRenamed().

- Use functions like upper(), when(), concat(), year(), current_date().

- Use groupBy() and agg() for aggregations.

- Explain join behavior differences (inner vs left).