
🧾 Dataset: Combine Existing Data

Reuse previous `employee_data` and `performance_data`, and now add a third:

📁 Dataset 3: `project_data`

```
project_data = [
    ("Ananya", "HR Portal", 120),
    ("Rahul", "Data Platform", 200),
    ("Priya", "Data Platform", 180),
    ("Zoya", "Campaign Tracker", 100),
    ("Karan", "HR Portal", 130),
    ("Naveen", "ML Pipeline", 220),
    ("Fatima", "Campaign Tracker", 90)
]
columns_proj = ["Name", "Project", "HoursWorked"]

df_proj = spark.createDataFrame(project_data, columns_proj)
```

🧠 PySpark Exercises – Set 3 (Project, Nulls, Functions)

🔗 Joins and Advanced Aggregations

1. Join `employee_data`, `performance_data`, and `project_data`.
2. Compute total hours worked per department.
3. Compute average rating per project.

🧹 Handling Missing Data (introduce some manually)

4. Add a row to `performance_data` with a `None` rating.
5. Filter rows with null values.
6. Replace null ratings with the department average.

🧮 Built-In Functions and UDF

7. Create a column `PerformanceCategory`:
    - Excellent (>=4.7),
    - Good (4.0–4.69),
    - Average (<4.0)
8. Create a UDF to assign bonus:
    - If project hours > 200 → ₹10,000
    - Else → ₹5,000

📅 Date and Time Functions

9. Add a column `JoinDate` with `2021-06-01` for all, then add `MonthsWorked` as difference from today.

10. Calculate how many employees joined before 2022.

Unions

11. Create another small team DataFrame and `union()` it with `employee_data`.

```
extra_employees = [
    ("Meena", "HR", 48000),
    ("Raj", "Marketing", 51000)
]
```

Saving Results

12. Save the final merged dataset (all 3 joins) as a partitioned Parquet file based on `Department`.
