
🆕 New Dataset: Employee Timesheet System

📄 Sample Data (Create as `employee_timesheet.csv`)

```
EmployeeID,Name,Department,Project,WorkHours,WorkDate,Location,Mode
E101,Anita,IT,Alpha,8,2024-05-01,Bangalore,Remote
E102,Raj,HR,Beta,7,2024-05-01,Mumbai,Onsite
E103,John,Finance,Alpha,5,2024-05-02,Delhi,Remote
E101,Anita,IT,Alpha,9,2024-05-03,Bangalore,Remote
E104,Meena,IT,Gamma,6,2024-05-03,Hyderabad,Onsite
E102,Raj,HR,Beta,8,2024-05-04,Mumbai,Remote
```

🧠 Task Set – Intermediate to Advanced PySpark (No DLT)

🔹 Data Ingestion & Schema Handling

1. Load the CSV using inferred schema.
2. Load the same file with schema explicitly defined.
3. Add a new column `Weekday` extracted from `WorkDate`.

🔹 Aggregations & Grouping

4. Calculate total work hours by employee.
5. Calculate average work hours per department.
6. Get top 2 employees by total hours using window function.

🔹 Date Operations

7. Filter entries where `WorkDate` falls on a weekend.
8. Calculate running total of hours per employee using window.

🔹 Joining DataFrames

9. Create `department_location.csv`:

```
Department,DeptHead
IT,Anand
HR,Shruti
Finance,Kamal
```

10. Join with timesheet data and list all employees with their DeptHead.

🔹 Pivot & Unpivot

11. Pivot table: total hours per employee per project.
12. Unpivot example: Convert mode-specific hours into rows.

🔹 UDF & Conditional Logic

13. Create a UDF to classify work hours:

```python
def workload_tag(hours):
    if hours >= 8: return "Full"
    elif hours >= 4: return "Partial"
    else: return "Light"
```

14. Add a column `WorkloadCategory` using this UDF.

🧹 Nulls and Cleanup

15. Introduce some nulls in Mode column.
16. Fill nulls with "Not Provided".
17. Drop rows where `WorkHours` < 4.

🧠 Advanced Conditions

18. Use `when-otherwise` to mark employees as "Remote Worker" if >80% entries are Remote.
19. Add a new column `ExtraHours` where hours > 8.

🔁 Union + Duplicate Handling

20. Append a dummy timesheet for new interns using `unionByName()`.
21. Remove duplicate rows based on all columns.
