Seekho Bigdata
Data is the New Oil

PySpark Scenario-Based Interview Questions

DAY 5 — Handling NULLs, Missing Data & Data Quality Checks

Karthik Kondpak
9989454737

PySpark Scenario-Based Interview Questions

DAY 5 — Handling NULLs, Missing Data & Data Quality Checks

Concepts Covered Today

- Identifying NULL values
- Handling NULLs using fillna and `dropna`
- Conditional NULL handling
- Data quality validation checks
- Real-time pipeline scenarios

Sample Data: `employee_df`

emp_id	emp_name	department	salary	joining_date
1	Rahul	IT	60000	2022-01-10
2	Priya	NULL	55000	2022-02-15
3	Amit	HR	NULL	2022-03-01
4	Neha	IT	65000	NULL

✅ Question 1: Identify Records Containing NULL Values

🔹 Scenario

Before analytics reporting, identify records that contain **any NULL values**.

🧪 PySpark Solution

```
null_records_df = employee_df.filter(
    employee_df.emp_name.isNull() |
    employee_df.department.isNull() |
    employee_df.salary.isNull() |
    employee_df.joining_date.isNull()
)

null_records_df.show()
```

📝 Explanation

- isNull() checks for missing values
- Common first step in data quality validation

✅ Question 2: Replace NULL Values with Default Values

🔹 Scenario

Replace missing values based on business rules:

- Department → "Unknown"
- Salary → 0

🧪 PySpark Solution

```
filled_df = employee_df.fillna({
"department": "Unknown",
"salary": 0
})

filled_df.show()
```

📝 Explanation

- `fillna()` is efficient for bulk NULL replacement

- Often used before aggregations

✅ Question 3: Drop Records with Critical NULL Values

🔹 Scenario

If `emp_name` or `j oining _date` is missing, the record should be **rejected**.

🧪 PySpark Solution

```
clean_df = employee_df.dropna(subset=["emp_name", "joining_date"])

clean_df.show()
```

📝 Explanation

- subset ensures only critical columns are validated
- Helps maintain data accuracy

✅ Question 4: Conditional NULL Handling Using when()

🔹 Scenario

If salary is NULL, assign default based on department:

- IT → 50000
- HR → 40000

🧪 PySpark Solution

```
from pyspark.sql.functions import when, col

conditional_df = employee_df.withColumn(
    "salary",
    when(col("salary").isNull() & (col("department")
== "IT"), 50000)
    .when(col("salary").isNull() & (col("department")
== "HR"), 40000)
    .otherwise(col("salary"))
)

conditional_df.show()
```

📝 Explanation

- Business-driven NULL handling
- Frequently discussed in real projects

✅ Question 5: Validate Data Quality Rules

🔹 Scenario

Apply data quality checks before loading into the warehouse:

- Salary must be > 0
- Joining date must not be NULL

🧪 PySpark Solution

```
invalid_records_df = employee_df.filter(
    (col("salary") <= 0) |
col("joining_date").isNull()
)

valid_records_df =
```

```
employee_df.subtract(invalid_records_df)

invalid_records_df.show()
valid_records_df.show()
```

📝 Explanation

- Invalid records can be redirected to error tables
- Demonstrates pipeline robustness

Let's build your Data Engineering journey together!

📩 Call us directly at: 9989454737

🌐 https://seekhobigdata.com/