



## Databricks – PySpark: When Otherwise (Case)

---

- ✓ When
- ✓ Otherwise
- ✓ Case

# Case Statement

To evaluate list of conditions and choose a result path according to the matching condition, `When().Otherwise()` function in Pyspark can be used

This is similar to case or switch statement in other programming languages

When no condition is matching, Otherwise result path would be chosen.



# Syntax

```
df.withColumn("New or existing column", when(Condition1, Result1)  
                                             .when(Condition2, Result2)  
                                             .when(ConditionN, ResultN)  
                                             .otherwise(Result))
```

```
df.withColumn("New or existing column ", expr("CASE WHEN Condition1 THEN Result1"+  
                                              "WHEN Condition2 THEN Result2"+  
                                              "WHEN ConditionN THEN ResultN" +  
                                              "ELSE Result END"))
```

To Combine multiple conditions: '&' for AND ; '|' for OR

## 12. Databricks | Pyspark: Case Function (When.Otherwise )

08 - Pyspark Switch Case (Python)

My Cluster | File | Edit | View: Standard | Permissions | Run All | Clear | Publish | Comments | Experiment | Revision history

Cmd 1

### Create sample Dataframe

Cmd 2

```
1 data_student = [("Raja", "Science", 80, "P", 90),
2                 ("Rakesh", "Maths", 90, "P", 70),
3                 ("Rama", "English", 20, "F", 80),
4                 ("Ramesh", "Science", 45, "F", 75),
5                 ("Rajesh", "Maths", 30, "F", 50),
6                 ("Raghav", "Maths", None, "NA", 70)]
7
8 Schema = ["name", "Subject", "Mark", "Status", "Attendance"]
9 df = spark.createDataFrame(data = data_student, schema = Schema)
10 display(df)
```

(3) Spark Jobs

df: pyspark.sql.dataframe.DataFrame = [name: string, Subject: string ... 3 more fields]

	name	Subject	Mark	Status	Attendance
1	Raja	Science	80	P	90
2	Rakesh	Maths	90	P	70
3	Rama	English	20	F	80
4	Ramesh	Science	45	F	75
5	Rajesh	Maths	30	F	50
6	Raghav	Maths	null	NA	70

Showing all 6 rows.

Google

08 - Pyspark Switch Case (Python)

My Cluster

File

Edit

View: Standard

Permissions

Run All

Clear

Publish

Comments

Experiment

Revision history

2	Rakesh	Maths	90	Pass	70
3	Rama	English	20	Fail	80
4	Ramesh	Science	45	Fail	75
5	Rajesh	Maths	30	Fail	50
6	Raghav	Maths	null	Absentee	70

Showing all 6 rows.

Command took 0.81 seconds -- by audaciousazure@gmail.com at 07/07/2021, 16:56:37 on My Cluster

Cmd 5

Create a new column

Cmd 6

```
1 from pyspark.sql.functions import when
2 df2 = df.withColumn("New_status", when(df.Mark >=50,"Pass")
3                                     .when(df.Mark <50,"Fail")
4                                     .otherwise("Absentee"))
5 |
6 display(df2)
```

(3) Spark Jobs

df2: pyspark.sql.dataframe.DataFrame = [name: string, Subject: string ... 4 more fields]

	name	Subject	Mark	Status	Attendance	New_status
1	Raja	Science	80	P	90	Pass
2	Rakesh	Maths	90	P	70	Pass



## 12. Databricks | Pyspark: Case Function (When.Otherwise )

Google

08 - Pyspark Switch Case (Python)

My Cluster | File | Edit | View: Standard | Permissions | Run All | Clear

Command took 0.56 seconds -- by audaciousazure@gmail.com at 07/07/2021, 16:57:36 on My Cluster

Cmd 7

### Another Syntax method

Cmd 8

```
1 from pyspark.sql.functions import expr
2
3
4 df3 = df.withColumn("new_status", expr("CASE WHEN Mark >= 50 THEN 'Pass' " +
5     "WHEN Mark < 50 THEN 'Fail' " +
6     "ELSE 'Absentee' END"))
7 display(df3)
```

Cmd 9

### Multi-conditions using AND and OR Operators

Cmd 10

```
1 from pyspark.sql.functions import when
2 df4 = df.withColumn("Grade", when((df.Mark >=80) & (df.Attendance >=80), "Distinction")
3     .when((df.Mark >=50) & (df.Attendance >=50), "Good")
4     .otherwise("Average"))
5
6 display(df4)
```

Shift+Enter to run

## 12. Databricks | Pyspark: Case Function (When.Otherwise )

08 - Pyspark Switch Case (Python)

My Cluster | File | Edit | View: Standard | Permissions | Run All | Clear | Publish | Comments | Experiment | Revision history

### Multi-conditions using AND and OR Operators

```
1 from pyspark.sql.functions import when
2 df4 = df.withColumn("Grade", when((df.Mark >=80) | (df.Attendance >=80), "Distinction")
3                               .when((df.Mark >=50) & (df.Attendance >=50), "Good")
4                               .otherwise("Average"))
5
6 display(df4)
```

(3) Spark Jobs

df4: pyspark.sql.dataframe.DataFrame = [name: string, Subject: string ... 4 more fields]

	name	Subject	Mark	Status	Attendance	Grade
1	Raja	Science	80	P	90	Distinction
2	Rakesh	Maths	90	P	70	Distinction
3	Rama	English	20	F	80	Distinction
4	Ramesh	Science	45	F	75	Average
5	Rajesh	Maths	30	F	50	Average
6	Raghav	Maths	null	NA	70	Average

Showing all 6 rows.

Command took 0.58 seconds -- by audaciousazure@gmail.com at 07/07/2021, 17:02:40 on My Cluster

Shift+Enter to run