

Experiment-12Q Column Expression

1 Import studentmarks_data.csv into Datalricks tables folder.

2 Import SQLContext

→ from pyspark.sql import SQLContext

3 Load the data and creating Dataframe and display 5 elements.

→ fileName = "dbs:/filestore/tables/student_marks_data.csv"

studentMarksDF = spark.read.csv(fileName, header="true", inferSchema="true")

studentMarksDF.show(5)

4 Print the schema using printSchema

→ studentMarksDF.printSchema()

5 Adding a New Column

→ from pyspark.sql.functions import exp
df_with_exp = studentMarksDF.withColumn("expText1", exp("+test1"))
df_with_exp.show(2).

6 Generating Random values in new column.

→ from pyspark.sql.functions import rand.
df_with_rand = studentMarksDF.withColumn
("rand", rand())
df_with_rand.show(2)

7 Calculating percentage of marks for every student.

→ df = studentMarksDF
df_with_percentage = df.withColumn("Percentage",
(df.test1 + df.test2 + df.test3 + df.test4)/4)
df_with_percentage.show(5)

8 Use of "when" statement and renaming the column.

→ from pyspark.sql import functions as F
K = df_with_percentage.select(df.Student_id, F.when(
df_with_percentage.Percentage > 50, "Pass").
.otherwise("Fail").alias("Result"))
K.show(10).

9 Sorting a column.

→ df_with_percentage.orderBy("Percentage").show(5)

10 Sort using SQL

→ df_with_percentage.createOrReplaceTempView
("percentage_table")

11 Execute sql query 'select * from percentage table ORDER BY Percentage' and display the result.

→ spark.sql ("select * from percentage table ORDER BY Percentage"). show()

12 Dropping a column.

→ dropDF = studentMarksDF.drop ("year_ID")

13 Print the schema

→ dropDF.printSchema()

14 Replacing contents of column and show 10 elements.

→ from pyspark.sql.functions import *
newDF = studentMarksDF.withColumn ("Subject_type",
regex_replace ("Subject_type", "PRC", "Practical")).
withColumn ("Subject_type", regex_replace (
"Subject_type", "THE", "Theory")).

newDF.show(10).

15 Fetching Column Object 'Subject Code' from DataFrame.

→ Column_SC = studentMarksDF ["Subject Code"]

16 Selecting Rows having Subject code 'PHY'

→ column_PHY = column_SC.endswith ("PHY")
studentMarksDF.select(column_PHY).show()

7. Similarly selecting the rows having subject type 'PRC'

```
→ column_ST = studentMarksDF['Subject_type']
column_PRC = column_ST.endswith("PRC")
studentMarksDF.select(column_PRC).show()
```

Outputs:

Loading data and creating DataFrame

```
+-----+-----+-----+-----+-----+-----+-----+
|Student_id|Subject_type|Subject_code|year_ID|test1|test2|test3|test4|
+-----+-----+-----+-----+-----+-----+-----+
| 10000001|      PRC|      CHE|  2014|  93|  34|  91|  10|
| 10000001|      PRC|      PHY|  2015|  42|  77|  62|  98|
| 10000001|      PRC|      PHY|  2016|  58|  68|  99|  82|
| 10000001|      THE|      PHY|  2015|  64|  64|  14|  27|
| 10000001|      THE|      CHE|  2014|  84|  98|  65|  99|
+-----+-----+-----+-----+-----+-----+-----+
only showing top 5 rows

Command took 12.29 seconds -- by 500068760@stu.upes.ac.in at 4/9/2021, 5:59:45 PM on sparkCluster
```

Printing schema using PrintSchema()

```
root
|-- Student_id: integer (nullable = true)
|-- Subject_type: string (nullable = true)
|-- Subject_code: string (nullable = true)
|-- year_ID: integer (nullable = true)
|-- test1: integer (nullable = true)
|-- test2: integer (nullable = true)
|-- test3: integer (nullable = true)
|-- test4: integer (nullable = true)

Command took 0.20 seconds -- by 500068760@stu.upes.ac.in at 4/9/2021, 5:59:45 PM on sparkCluster
```


Adding a New Column

```
▶ (1) Spark Jobs
▶ df_with_exp: pyspark.sql.dataframe.DataFrame = [Student_id: integer, Subject_type: string ... 7 more fields]
+-----+-----+-----+-----+-----+-----+-----+-----+
|Student_id|Subject_type|Subject_code|year_ID|test1|test2|test3|test4|expTest1|
+-----+-----+-----+-----+-----+-----+-----+-----+
| 10000001|      PRC|      CHE|  2014|  93|  34|  91|  10|2.451245542920086E40|
| 10000001|      PRC|      PHY|  2015|  42|  77|  62|  98|1.739274941520500...|
+-----+-----+-----+-----+-----+-----+-----+-----+
only showing top 2 rows

Command took 1.38 seconds -- by 500068760@stu.upes.ac.in at 4/9/2021, 5:59:45 PM on sparkCluster
```

Generating random values

▶ (1) Spark Jobs

▶  df_with_rand: pyspark.sql.dataframe.DataFrame = [Student_id: integer, Subject_type: string ... 7 more fields]


```
+-----+-----+-----+-----+-----+-----+-----+-----+
|Student_id|Subject_type|Subject_code|year_ID|test1|test2|test3|test4|rand()|
+-----+-----+-----+-----+-----+-----+-----+-----+
| 10000001|PRC|CHE|2014|93|34|91|10|0.06446953656048315|
| 10000001|PRC|PHY|2015|42|77|62|98|0.5148722488111165|
+-----+-----+-----+-----+-----+-----+-----+-----+
```


only showing top 2 rows

Command took 0.83 seconds -- by 500068760@stu.upes.ac.in at 4/9/2021, 5:59:45 PM on sparkCluster

Calculating percentage of marks of every student

▶ (1) Spark Jobs

▶  df: pyspark.sql.dataframe.DataFrame = [Student_id: integer, Subject_type: string ... 6 more fields]

▶  df_with_percentage: pyspark.sql.dataframe.DataFrame = [Student_id: integer, Subject_type: string ... 7 more fields]

```
+-----+-----+-----+-----+-----+-----+-----+-----+
|Student_id|Subject_type|Subject_code|year_ID|test1|test2|test3|test4|Percentage|
+-----+-----+-----+-----+-----+-----+-----+-----+
| 10000001|PRC|CHE|2014|93|34|91|10|57.0|
| 10000001|PRC|PHY|2015|42|77|62|98|69.75|
| 10000001|PRC|PHY|2016|58|68|99|82|76.75|
| 10000001|THE|PHY|2015|64|64|14|27|42.25|
| 10000001|THE|CHE|2014|84|98|65|99|86.5|
+-----+-----+-----+-----+-----+-----+-----+-----+
```

only showing top 5 rows

Using when statement

▶ (1) Spark Jobs

▶  k: pyspark.sql.dataframe.DataFrame = [Student_id: integer, Result: string]

```
+-----+-----+
|Student_id|Result|
+-----+-----+
| 10000001|Pass|
| 10000001|Pass|
| 10000001|Pass|
| 10000001|Fail|
| 10000001|Pass|
| 10000001|Pass|
| 10000001|Fail|
| 10000001|Pass|
| 10000001|Fail|
| 10000001|Fail|
+-----+-----+
```

only showing top 10 rows

Sorting a column

```
1 df_with_percentage.orderBy('Percentage').show(5)
```

► (1) Spark Jobs

Student_id	Subject_type	Subject_code	year_ID	test1	test2	test3	test4	Percentage
10000041	THE	CHE	2016	15	10	14	16	13.75
10000887	THE	PHY	2015	10	11	13	28	15.5
10000187	THE	CHE	2016	17	21	11	17	16.5
10000022	THE	BIO	2015	18	10	22	17	16.75
10000403	PRC	CHE	2014	15	32	11	12	17.5

only showing top 5 rows

Command took 1.25 seconds -- by 500068760@stu.upes.ac.in at 4/9/2021, 5:59:45 PM on sparkCluster

Executing SQL query

► (1) Spark Jobs

Student_id	Subject_type	Subject_code	year_ID	test1	test2	test3	test4	Percentage
10000041	THE	CHE	2016	15	10	14	16	13.75
10000887	THE	PHY	2015	10	11	13	28	15.5
10000187	THE	CHE	2016	17	21	11	17	16.5
10000022	THE	BIO	2015	18	10	22	17	16.75
10000403	PRC	CHE	2014	15	32	11	12	17.5
10000435	PRC	BIO	2016	20	14	22	16	18.0
10000815	THE	BIO	2014	16	25	21	12	18.5
10000464	PRC	BIO	2016	16	26	15	17	18.5
10000421	THE	PHY	2015	19	17	18	21	18.75
10000945	THE	CHE	2016	30	13	15	18	19.0
10000219	THE	CHE	2015	33	18	12	13	19.0
10000311	PRC	CHE	2014	16	10	11	39	19.0
10000162	PRC	CHE	2014	11	10	36	20	19.25
10000498	PRC	SCI	2014	30	23	10	14	19.25
10000050	PRC	BIO	2016	17	10	20	30	19.25
10000942	THE	PHY	2014	35	10	12	20	19.25
10000226	PRC	CHE	2016	22	10	13	33	19.5
10000131	THE	CHE	2016	34	17	14	14	19.75
10000040	THE	PHY	2014	14	20	24	12	20.0

Command took 1.29 seconds -- by 500068760@stu.upes.ac.in at 4/9/2021, 5:59:45 PM on sparkCluster

Printing schema after drop function

```
root
|-- Student_id: integer (nullable = true)
|-- Subject_type: string (nullable = true)
|-- Subject_code: string (nullable = true)
|-- test1: integer (nullable = true)
|-- test2: integer (nullable = true)
|-- test3: integer (nullable = true)
|-- test4: integer (nullable = true)
```

Command took 0.04 seconds -- by 500068760@stu.upes.ac.in at 4/9/2021, 5:59:45 PM on sparkCluster

Replacing contents of column

```
▶ (1) Spark Jobs
▶ newDF: pyspark.sql.dataframe.DataFrame = [Student_id: integer, Subject_type: string ... 6 more fields]
+-----+-----+-----+-----+-----+-----+-----+
|Student_id|Subject_type|Subject_code|year_ID|test1|test2|test3|test4|
+-----+-----+-----+-----+-----+-----+-----+
| 10000001| Practical|      CHE|  2014|  93|  34|  91|  10|
| 10000001| Practical|      PHY|  2015|  42|  77|  62|  98|
| 10000001| Practical|      PHY|  2016|  58|  68|  99|  82|
| 10000001|  Theory|      PHY|  2015|  64|  64|  14|  27|
| 10000001|  Theory|      CHE|  2014|  84|  98|  65|  99|
| 10000001|  Theory|      SCI|  2016|  62|  21|  93|  25|
| 10000001| Practical|      SCI|  2015|  21|  28|  60|  74|
| 10000001|  Theory|      CHE|  2014|  77|  29|  81|  62|
| 10000001|  Theory|      BIO|  2015|  17|  37|  21|  37|
| 10000001|  Theory|      PHY|  2015|  89|  23|  11|  30|
+-----+-----+-----+-----+-----+-----+-----+
only showing top 10 rows
```

Selecting rows having Sub_Code as PHY

```
▶ (1) Spark Jobs
+-----+
|endswith(Subject_code, PHY)|
+-----+
|
|      false|
|      true|
|      true|
|      true|
|      false|
|      false|
|      false|
|      false|
|      false|
|      true|
|      true|
|      false|
|      false|
|      true|
|      false|
|      true|
|      false|
|      true|
|      false|
+-----+
Command took 0.51 seconds -- by 500068760@stu.upes.ac.in at 4/9/2021, 5:59:45 PM on sparkCluster
```


Selecting rows having Sub_type as PRC

► (1) Spark Jobs

```
+-----+
|endswith(Subject_type, PRC)|
+-----+
|          true|
|          true|
|          true|
|         false|
|         false|
|         false|
|          true|
|         false|
|         false|
|         false|
|         false|
|         false|
|         false|
|         false|
|         false|
|         false|
|         false|
|          true|
|          true|
|          true|
|         false|
```

Command took 0.41 seconds -- by 500068760@stu.upes.ac.in at 4/9/2021, 5:59:46 PM on sparkCluster