**Akilesh K**

k.akilesh123@gmail.com

**Data engineering - Batch 1**

**Date: 10-02-24**

# DAY 15 - PYSPARK – Views

## Create spark session

```
1  from pyspark.sql import SparkSession
2  spark=SparkSession.builder.appName("Practice").config("spark.sql.catalogImplementation","hive").getOrCreate()
```

```
Command took 0.20 seconds -- by kakilesh123@gmail.com at 2/10/2024, 2:42:15 PM on hexaCluster
```

## Create database

```
1    spark.sql("CREATE DATABASE customer_db;")
```

```
ıt[3]: DataFrame[]

ımmand took 6.49 seconds -- by kakilesh123@gmail.com at 2/10/2024, 2
```

## Add value if not exist

```
Cmd 38

1   spark.sql("CREATE DATABASE IF NOT EXISTS customer_db COMMENT 'This is customer database'WITH DBPROPERTIES (ID=1, Name='John')");

Command took 0.27 seconds -- by kakilesh123@gmail.com at 2/10/2024, 2:43:00 PM on hexaCluster

Cmd 39
```

## Describe database

```
Cmd 39

    1    spark.sql("DESCRIBE DATABASE EXTENDED customer_db;")

Out[5]: DataFrame[database_description_item: string, database_description_value: string]

Command took 1.34 seconds -- by kakilesh123@gmail.com at 2/10/2024, 2:44:24 PM on hexaCluster
```

## Spark session

Cmd 40

```
1  from pyspark.sql import SparkSession
2
3  spark = SparkSession \
4      .builder \
5      .appName("Python Spark SQL basic example") \
6      .config("spark.some.config.option", "some-value") \
7      .getOrCreate()
```

Command took 0.06 seconds -- by kakilesh123@gmail.com at 2/10/2024, 3:04:14 PM on hexaCluster

## Create dataframe

```
1  from pyspark.sql.functions import col
2  data = [(None, "Michael"), (30, "Andy"), (19, "Justin")]
3  df = spark.createDataFrame(data, schema=["age", "name"])
4  df.show()
```

▸ (3) Spark Jobs

▸ df: pyspark.sql.dataframe.DataFrame = [age: long, name: string]

```
+----+-------+
| age|   name|
+----+-------+
|null|Michael|
|  30|   Andy|
|  19| Justin|
+----+-------+
```

Command took 7.83 seconds -- by kakilesh123@gmail.com at 2/10/2024, 3:09:15 PM on hexaCluster

## Adding 1 to a row

```
df.printSchema()
df.select("name").show()
df.select(df['name'], df['age'] + 1).show()

```

▸ (6) Spark Jobs

```
root
 |-- age: long (nullable = true)
 |-- name: string (nullable = true)

+-------+
|   name|
+-------+
|Michael|
|   Andy|
| Justin|
+-------+

+-------+----------+
|   name|(age + 1)|
+-------+----------+
|Michael|      null|
|   Andy|        31|
| Justin|        20|
+-------+----------+
```

## Filter and group by

```
1  df.filter(df['age'] > 21).show()
2  df.groupBy("age").count().show()
3
```

▸ (5) Spark Jobs

```
+---+----+
|age|name|
+---+----+
| 30|Andy|
+---+----+

+----+-----+
| age|count|
+----+-----+
|null|    1|
|  30|    1|
|  19|    1|
+----+-----+
```

Command took 4.80 seconds -- by kakilesh123@gmail.com at 2/10/2024, 3:10:40 PM on hexaCluster

**Views using sparksql**

```
1  df.createOrReplaceTempView("people")
2  sqlDF = spark.sql("SELECT * FROM people")
3  sqlDF.show()
```

▸ (3) Spark Jobs

▸ sqlDF: pyspark.sql.dataframe.DataFrame = [age: long, name: string]

```
+----+-------+
| age|   name|
+----+-------+
|null|Michael|
|  30|   Andy|
|  19| Justin|
+----+-------+
```

Command took 0.96 seconds -- by kakilesh123@gmail.com at 2/10/2024, 3:11:10 PM on hexaCluster

## Global view

```
1  df.createGlobalTempView("people")
2  spark.sql("SELECT * FROM global_temp.people").show()
3  spark.newSession().sql("SELECT * FROM global_temp.people").show()
4
```

▸ (6) Spark Jobs

```
+----+-------+
| age|   name|
+----+-------+
|null|Michael|
|  30|   Andy|
|  19| Justin|
+----+-------+

+----+-------+
| age|   name|
+----+-------+
|null|Michael|
|  30|   Andy|
|  19| Justin|
+----+-------+
```

Command took 1.29 seconds -- by kakilesh123@gmail.com at 2/10/2024, 3:11:51 PM on hexaCluster