Spark SQL G1 and G2 Notebook 2023-08-28 12:31:23

(https://databricks.com)

```
val sqlConstext = new org.apache.spark.sql.SQLContext(sc)
```

```
command-4344394284453045:1: warning: constructor SQLContext in class SQLContext is deprecated (since 2.0.0): Use SparkSession.builde
r instead
val sqlConstext = new org.apache.spark.sql.SQLContext(sc)
                  ^
sqlConstext: org.apache.spark.sql.SQLContext = org.apache.spark.sql.SQLContext@508225e8
```

```
val a  = sc.parallelize(1 to 10)
```

```
a: org.apache.spark.rdd.RDD[Int] = ParallelCollectionRDD[17] at parallelize at command-4344394284453046:1
```

```
val b = a.map(x=> (x , x+1))
```

```
b: org.apache.spark.rdd.RDD[(Int, Int)] = MapPartitionsRDD[18] at map at command-4344394284453047:1
```

```
b.collect
```

```
res0: Array[(Int, Int)] = Array((1,2), (2,3), (3,4), (4,5), (5,6), (6,7), (7,8), (8,9), (9,10), (10,11))
```

```
val df = b.toDF("First", "Second")
```

```
df: org.apache.spark.sql.DataFrame = [First: int, Second: int]
```

```
df.show
```

```
+-----+------+
|First|Second|
+-----+------+
|    1|     2|
|    2|     3|
|    3|     4|
|    4|     5|
|    5|     6|
|    6|     7|
|    7|     8|
|    8|     9|
|    9|    10|
|   10|    11|
+-----+------+
```

```
val a  = List(("Tom", 5),("Jerry", 2),("Donald", 7))
```

```
a: List[(String, Int)] = List((Tom,5), (Jerry,2), (Donald,7))
```

```
val df = a.toDF("Name", "Age")
```

```
df: org.apache.spark.sql.DataFrame = [Name: string, Age: int]
```

```
df.show
```

```
+------+---+
|  Name|Age|
+------+---+
|   Tom|  5|
| Jerry|  2|
|Donald|  7|
+------+---+
```

```
val a  = Seq(("Tom", 5),("Jerry", 2),("Donald", 7))
```

```
a: Seq[(String, Int)] = List((Tom,5), (Jerry,2), (Donald,7))
```

```
val df = a.toDF("Name", "Age")
```

```
df: org.apache.spark.sql.DataFrame = [Name: string, Age: int]
```

```
df.show
```

```
+------+---+
|  Name|Age|
+------+---+
|   Tom|  5|
| Jerry|  2|
|Donald|  7|
+------+---+
```

```
df.registerTempTable("Cartoon")
```

```
command-4344394284453057:1: warning: method registerTempTable in class Dataset is deprecated (since 2.0.0): Use createOrReplaceTempV
iew(viewName) instead.
df.registerTempTable("Cartoon")
   ^
```

```
df.createOrReplaceTempView("Cartoon")
```

```
sqlContext.sql("select * from Cartoon where Name = 'Tom'").show
```

```
+----+---+
|Name|Age|
+----+---+
| Tom|  5|
+----+---+
```

```
sqlContext.sql("select * from Cartoon").show
```

```
+------+---+
|  Name|Age|
+------+---+
|   Tom|  5|
| Jerry|  2|
|Donald|  7|
+------+---+
sqlContext.sql("select count(*) from Cartoon").show
```

```
+--------+
|count(1)|
+--------+
|       3|
+--------+
```

```
// questions : to create a json file, upoad it open dbfs and perform the following operations on it.

// printSchema()
// select the query with all the names
// filter and identify age > 23
// groupBy Age Count it and show it

// how ro read file

// var df1 = spark.read.format("json").load("dbfs:/FileStore/shared_uploads/........./........json")
```

```
var df1 = spark.read.format("json").load("/FileStore/tables/file.json")
```

```
df1: org.apache.spark.sql.DataFrame = [_corrupt_record: string]
```

```
display(df1)
```

```
   AnalysisException: Since Spark 2.3, the queries from raw JSON/CSV files are disallowed when the
referenced columns only include the internal corrupt record column
(named _corrupt_record by default). For example:
spark.read.schema(schema).csv(file).filter($"_corrupt_record".isNotNull).count()
and spark.read.schema(schema).csv(file).select("_corrupt_record").show().
Instead, you can cache or save the parsed results and then send the same query.
For example, val df = spark.read.schema(schema).csv(file).cache() and then
df.filter($"_corrupt_record".isNotNull).count().
```

```
val df1 = spark.read.format("json").load("dbfs:/FileStore/shared_uploads/devjethva234@gmail.com/emp_1.json")
```

```
df1: org.apache.spark.sql.DataFrame = [age: string, id: string ... 1 more field]
```

```
df1.show
```

```
+---+----+---------+
|age|  id|     name|
+---+----+---------+
| 25|1201|       om|
| 25|1202|     some|
| 25|1203|    thing|
| 25|1204|different|
| 25|1205|    going|
| 25|1206|       on|
| 25|1207|    kavan|
+---+----+---------+
```

```
   AnalysisException: [TABLE_OR_VIEW_NOT_FOUND] The table or view `df1` cannot be found. Verify the spelling and correctness of the s
chema and catalog.
If you did not qualify the name with a schema, verify the current_schema() output, or qualify the name with the correct schema and ca
```

```
talog.
To tolerate the error on drop use DROP VIEW IF EXISTS or DROP TABLE IF EXISTS.; line 1 pos 17;
'Project ['name]
+- 'UnresolvedRelation [df1], [], false
```