1. Exploring DataFrames using the Apache Spark Shell – Scala – 35 Minutes

Following features of Spark will be demonstrated here:

- Loading json file.
- Understand its schema
- Select required fields.
- Apply filter.

Start spark-shell

#spark-shell
Create a text file users.json which contains sample data as listed below in data folder:

```
{"name":"Alice", "pcode":"94304"}
{"name":"Brayden", "age":30,
"pcode":"94304"}
{"name":"Carla", "age":19,
"pcode":"10036"}
{"name":"Diana", "age":46}
```

Scala:

Initiate the spark-shell from the folder which you have created the above file.

```
// Read the users json file as a dataframe.
val usersDF = spark.read.json("users.json")

// Find out the schema of the uploaded file
usersDF.printSchema()
```

As shown above, three fields will be displayed according to the json fields specified in the text file.

```
scala> val usersDF = spark.read.json("users.json")
usersDF: org.apache.spark.sql.DataFrame = [age: bigint, name: string ... 1 more field]

scala> usersDF.printSchema
root
 |-- age: long (nullable = true)
 |-- name: string (nullable = true)
 |-- pcode: string (nullable = true)


scala>
```

//Let us find out the first 3 records to have a sample data.
val users = usersDF.take(3)
usersDF.show()

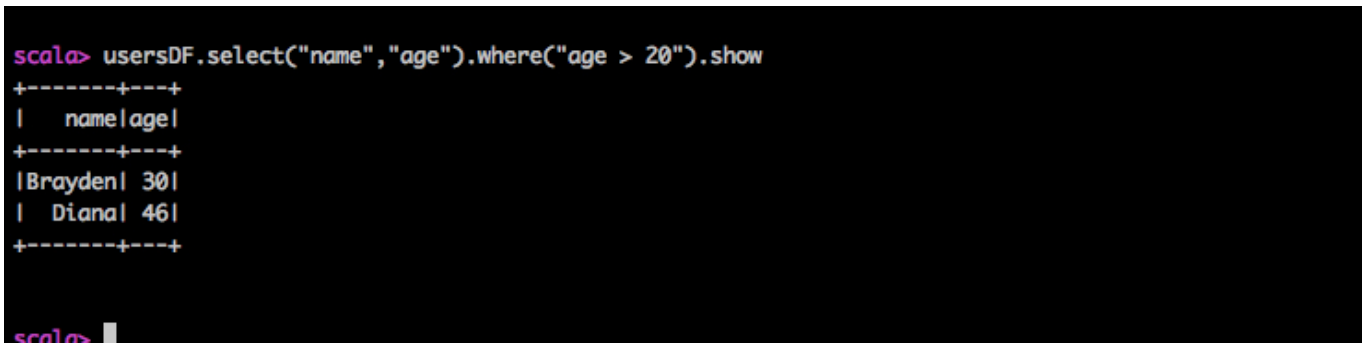
```
scala> val users = usersDF.take(3)
users: Array[org.apache.spark.sql.Row] = Array([null,Alice,94304], [30,Brayden,94304], [19,Carla,10036])

scala> usersDF.show()
+----+-------+-----+
| age|   name|pcode|
+----+-------+-----+
|null|  Alice|94304|
|  30|Brayden|94304|
|  19|  Carla|10036|
|  46|  Diana| null|
|null|Etienne|94104|
+----+-------+-----+


scala>
```

Out of the three fields, we are interested in only name and age fields. So, let us create a dataframe with only these two fields and apply a filter expression in which only person greater than 20 years are there in the dataframe.

val nameAgeDF = usersDF.select("name","age")
val nameAgeOver20DF = nameAgeDF.where("age > 20")
nameAgeOver20DF.show


scala> val nameAgeDF = usersDF.select("name","age")
nameAgeDF: org.apache.spark.sql.DataFrame = [name: string, age: bigint]
scala> val nameAgeOver20DF = nameAgeDF.where("age > 20")
nameAgeOver20DF: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [name: string, age: bigint]
scala> nameAgeOver20DF.show
+-------+---+
| name|age|
+-------+---+
|Brayden| 30|
| Diana| 46|
+-------+---+
scala>

usersDF.select("name","age").where("age > 20").show



You can also combine the functions as shown above. You will get the same result.

---------- Lab Ends Here----------
